
VVV VVV MMM MMM SSSSSSSSSSS RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
VVV VVV MMM MMM SSSSSSSSSSS RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
VVV VVV MMM MMM SSSSSSSSSSS RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
VVV VVV MMMMMM MMMMMM SSS RRR RRR TTT LLL
VVV VVV MMMMMM MMMMMM SSS RRR RRR TTT LLL
VVV VVV MMMMMM MMMMMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSSSSSSSS RRRRRRRRRRR TTT LLL
VVV VVV MMM MMM SSSSSSSSS RRRRRRRRRRR TTT LLL
VVV VVV MMM MMM SSSSSSSSS RRRRRRRRRRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV VVV MMM MMM SSS RRR RRR TTT LLL
VVV MMM MMM SSSSSSSSSSS RRR RRR TTT LLLLLLLLLLLLLLL
VVV MMM MMM SSSSSSSSSSS RRR RRR TTT LLLLLLLLLLLLLLL
VVV MMM MMM SSSSSSSSSSS RRR RRR TTT LLLLLLLLLLLLLLL

FILEID**VMSVECTOR

```
VV      VV  MM      MM   SSSSSSSS  VV      VV  EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT    OOOOOO    RRRRRRRR
VV      VV  MM      MM   SSSSSSSS  VV      VV  EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT    OOOOOO    RRRRRRRR
VV      VV  MMMM  MMMM  SS         VV      VV  EE           CC            TT        OO      OO  RR      RR
VV      VV  MMMM  MMMM  SS         VV      VV  EE           CC            TT        OO      OO  RR      RR
VV      VV  MM  MM  MM  SS         VV      VV  EE           CC            TT        OO      OO  RR      RR
VV      VV  MM  MM  MM  SS         VV      VV  EE           CC            TT        OO      OO  RR      RR
VV      VV  MM      MM   SSSSSS    VV      VV  EEEEEEEE     CC            TT        OO      OO  RRRRRRRR
VV      VV  MM      MM   SSSSSS    VV      VV  EEEEEEEE     CC            TT        OO      OO  RRRRRRRR
VV      VV  MM      MM         SS  VV      VV  EE           CC            TT        OO      OO  RR  RR
VV      VV  MM      MM         SS  VV      VV  EE           CC            TT        OO      OO  RR  RR
  VV  VV    MM      MM         SS    VV  VV    EE           CC            TT        OO      OO  RR    RR      ....
  VV  VV    MM      MM         SS    VV  VV    EE           CC            TT        OO      OO  RR    RR      ....
    VV      MM      MM  SSSSSSSS      VV      EEEEEEEEEE    CCCCCCCC      TT          OOOOOO    RR      RR    ....
    VV      MM      MM  SSSSSSSS      VV      EEEEEEEEEE    CCCCCCCC      TT          OOOOOO    RR      RR    ....
```

```
LL            IIIIII     SSSSSSSS
LL            IIIIII     SSSSSSSS
LL              II     SS
LL              II     SS
LL              II     SS
LL              II     SS
LL              II       SSSSSS
LL              II       SSSSSS
LL              II             SS
LL              II             SS
LL              II             SS
LL              II             SS
LLLLLLLLLL    IIIIII     SSSSSSSS
LLLLLLLLLL    IIIIII     SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  VMS$VECTOR - Define entry vectors for VMSRTL
0000     2          .IDENT  /4-003/                 ; File: VMSVECTOR.MAR  Edit: MDL4003
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28 ;
0000    29 ; FACILITY: VAX/VMS Run-Time Library
0000    30 ;++
0000    31 ; ABSTRACT:
0000    32 ;       This module contains the entry vector for the shareable image
0000    33 ;       VMSRTL.EXE.  VMSRTL is now only a "stub" that references procedures
0000    34 ;       in LIBRTL, MTHRTL, BASRTL, COBRTL and FORRTL.
0000    35 ;
0000    36 ;--
0000    37 ;
0000    38 ; VERSION: 1
0000    39 ;
0000    40 ; Revision History:
0000    41 ;
0000    42 ;******
0000    43 ;*
0000    44 ;*  WARNING!!!
0000    45 ;*
0000    46 ;*  The order or contents of the VMSRTL vector must never change!
0000    47 ;*
0000    48 ;******
0000    49 ;
0000    50 ; 4-001 - Modified from ALLGBL.MAR to only produce vector declarations.
0000    51 ;         SBL 11-May-1983
0000    52 ; 4-002 - Add MTH$AB_ALOG and MTH$AB_ATAN table copies to end.  SBL 20-May-1983
0000    53 ; 4-003 - Add OLDENTRY macro for obsolete entry points.  MDL 26-Sep-1983
0000    54 ;--
```

```
0000     56 ;+
0000     57 ; NOTE: This module contains many comments which are now of only historical
0000     58 ;       significance.  The image VMSRTL mostly consists of vectored entry
0000     59 ;       points that refer to procedures in other shareable images.  However,
0000     60 ;       a few data tables that were in VMSRTL remain since they cannot be
0000     61 ;       revectored.
0000     62 ;-
0000     63
0000     64 ;+
0000     65 ; Define macro MAC to generate vector entries.
0000     66 ;
0000     67 ; call: MAC     VEC_TYPE, VEC_AREA, SYMBOL, MASK
0000     68 ;
0000     69 ; where VEC_TYPE is:     CALL     - call entry point transfer vector
0000     70 ;                       JSB      - JSB entry point transfer vector
0000     71 ;                       NOVECT   - do not have a transfer vector
0000     72 ;                       SYM      - this is a symbol, not an entry point
0000     73 ;                       DATA     - this is data, kept in the vector
0000     74 ;                       FUTURE   - this is a proposed entry point, not yet
0000     75 ;                                  implemented, but space reserved.
0000     76 ;       VEC_AREA is:    FOR      - FORTRAN entry points
0000     77 ;                       LIB      - library entry points
0000     78 ;                       MTH      - Math library entry points
0000     79 ;                       STR      - String library entry points
0000     80 ;                       OTS      - Language independent entry points
0000     81 ;                       BAS      - BASIC-PLUS-2 entry points
0000     82 ;                       COB      - COBOL
0000     83 ;
0000     84 ;                       Note: VEC_AREA is ignored
0000     85 ;
0000     86 ;       SYMBOL is:      any entry point symbol
0000     87 ;       MASK is:        optional entry mask if not same as SYMBOL
0000     88 ;
0000     89 ;       Each entry vector is 8 bytes long and contains a 2 byte mask and
0000     90 ;       a 6 byte JMP instruction (for CALLs) or
0000     91 ;       a 6 byte JMP plus 2 filler bytes for JSBs.
0000     92 ;
0000     93 ;-
0000     94
```

```
                     96 .MACRO  MAC     VEC_TYPE, VEC_AREA, SYMBOL, MASK
                     97     .IF IDN     VEC_TYPE, JSB
                     98 $$'SYMBOL'::
                     99         JMP     G^SYMBOL                ; branch to JSB routine
                    100         .BYTE   0,0                     ; fill out to 8 bytes
                    101     .ENDC
                    102
                    103     .IF IDN     VEC_TYPE, CALL
                    104 $$'SYMBOL'::
                    105         .IF B MASK
                    106             .MASK   SYMBOL
                    107         .IFF
                    108             .MASK   MASK                ; get mask from other name
                    109         .ENDC
                    110             JMP     G^SYMBOL+2          ; branch to CALL+2 routine
                    111     .ENDC
                    112
                    113     .IF IDN     VEC_TYPE, FUTURE        ; Reserve space for future vector?
                    114         .BYTE   0,0,0,0,0,0,0,0         ; leave 8 bytes
                    115     .ENDC
                    116
                    117     .IF IDN     VEC_TYPE, DATA
                    118 $$'SYMBOL'_V::
                    119         .ADDRESS        SYMBOL-.        ; from non-shared routine.  Has format:
                    120         .BLKL           1               ;
                    121                                         ;    .ADDRESS   table_name-.
                    122         .ENDC                           ;    .BLKL      1
                    123
                    124 .ENDM
                    125
                    126 .MACRO  OLDENTRY        SYMBOL
                    127 $$'SYMBOL'::
                    128 .ENDM
                    129
00000000            130         .PSECT  $VMS$VECTOR PIC,USR,CON,REL,LCL,SHR,EXE,RD,NOWRT,PAGE
                    131 RTL$START:
                    132
```

```
0000   134 ;+
0000   135 ; FORTRAN compatibility routines - do not VECTOR
0000   136 ;-
0000   137
0000   138
0000   139 ; MODULE:COM$ASSIGN
0000   140         MAC     NOVECT  COM     ASSIGN
0000   141
0000   142 ; MODULE:COM$CLOSE
0000   143         MAC     NOVECT  COM     CLOSE
0000   144
0000   145 ; MODULE:COM$ERRSET
0000   146         MAC     NOVECT  COM     ERRSET
0000   147
0000   148 ; MODULE:COM$ERRTST
0000   149         MAC     NOVECT  COM     ERRTST
0000   150
0000   151 ; MODULE:COM$FDBSET
0000   152         MAC     NOVECT  COM     FDBSET
0000   153
0000   154 ; MODULE:COM$IRAD50
0000   155         MAC     NOVECT  COM     IRAD50
0000   156
0000   157 ; MODULE:COM$R50ASC
0000   158         MAC     NOVECT  COM     R50ASC
0000   159
0000   160 ; MODULE:COM$RAD50
0000   161         MAC     NOVECT  COM     RAD50
0000   162
0000   163 ; MODULE:COM$USEREX
0000   164         MAC     NOVECT  COM     USEREX
```

```
                        ;+
                        ; FORTRAN entry points
                        ; Put most frequently used FORTRAN entry points together first,
                        ; ie. I/O and OPEN and CLOSE.
                        ;-

                        ; MODULE:FOR$CLOSE
                                MAC     CALL    FOR     FOR$CLOSE
                        ; MODULE:FOR$ENTRY
                                MAC     CALL    FOR     FOR$DECODE_MF   FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$DECODE_MO   FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$ENCODE_MF   FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$ENCODE_MO   FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$READ_KF     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_KO     FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$READ_DF     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_DO     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_DU     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_SF     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_SL     FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$READ_SO     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$READ_SU     FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$WRITE_DF    FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$WRITE_DO    FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$WRITE_DU    FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$WRITE_SF    FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$WRITE_SL    FOR$$IO_BEG
                                MAC     CALL    FOR     FOR$WRITE_SO    FOR$$IO_BEG

                                MAC     CALL    FOR     FOR$WRITE_SU    FOR$$IO_BEG

                        ; MODULE:FOR$IO_END
                                MAC     CALL    FOR     FOR$IO_END

                        ; MODULE:FOR$IO_ELEM

                                MAC     CALL    FOR     FOR$IO_F_R
                                MAC     CALL    FOR     FOR$IO_F_V
                                MAC     CALL    FOR     FOR$IO_D_R
                                MAC     CALL    FOR     FOR$IO_D_V

                                MAC     CALL    FOR     FOR$IO_L_R
                                MAC     CALL    FOR     FOR$IO_L_V
                                MAC     CALL    FOR     FOR$IO_B_R
                                MAC     CALL    FOR     FOR$IO_B_V

                                MAC     CALL    FOR     FOR$IO_T_DS

                                MAC     CALL    FOR     FOR$IO_W_R
                                MAC     CALL    FOR     FOR$IO_W_V
                                MAC     CALL    FOR     FOR$IO_G_R
                                MAC     CALL    FOR     FOR$IO_G_V
```

```
0118   223          MAC     CALL    FOR     FOR$IO_H_R
0120   224          MAC     CALL    FOR     FOR$IO_H_V
0128   225
0128   226          MAC     CALL    FOR     FOR$IO_DC_R
0130   227          MAC     CALL    FOR     FOR$IO_GC_R
0138   228 ;                                by value calls at end
0138   229          MAC     CALL    FOR     FOR$IO_T_V_DS
0140   230          MAC     CALL    FOR     FOR$IO_FC_R
0148   231          MAC     CALL    FOR     FOR$IO_FC_V
0150   232          MAC     CALL    FOR     FOR$IO_LU_R
0158   233          MAC     CALL    FOR     FOR$IO_LU_V
0160   234          MAC     CALL    FOR     FOR$IO_WU_R
0168   235          MAC     CALL    FOR     FOR$IO_WU_V
0170   236          MAC     CALL    FOR     FOR$IO_X_DA
0178   237
0178   238 ; MODULE:FOR$OPEN
0178   239          MAC     CALL    FOR     FOR$OPEN
0180   240
0180   241 ;+
0180   242 ; Rest of FOR$ entries alphabetical order
0180   243 ;-
0180   244
0180   245 ; MODULE:FOR$BACKSPACE
0180   246          MAC     CALL    FOR     FOR$BACKSPACE
0188   247
0188   248 ; MODULE:FOR$BITOPS
0188   249          MAC     NOVECT  FOR     FOR$IMVBITS
0188   250          MAC     NOVECT  FOR     FOR$JMVBITS
0188   251          MAC     NOVECT  FOR     FOR$IIBITS
0188   252          MAC     NOVECT  FOR     FOR$JIBITS
0188   253          MAC     NOVECT  FOR     FOR$IISHFTC
0188   254          MAC     NOVECT  FOR     FOR$JISHFTC
0188   255          MAC     NOVECT  FOR     FOR$BITEST
0188   256          MAC     NOVECT  FOR     FOR$BJTEST
0188   257          MAC     NOVECT  FOR     FOR$IIBSET
0188   258          MAC     NOVECT  FOR     FOR$JIBSET
0188   259          MAC     NOVECT  FOR     FOR$IIBCLR
0188   260          MAC     NOVECT  FOR     FOR$JIBCLR
0188   261
0188   262 ; MODULE:OTS$CVTLT                ; New entry points at end
0188   263          MAC     CALL    FOR     FOR$CNV_OUT_I
0190   264          MAC     CALL    FOR     FOR$CNV_OUT_L
0198   265          MAC     CALL    FOR     FOR$CNV_OUT_O
01A0   266          MAC     CALL    FOR     FOR$CNV_OUT_Z
01A8   267
01A8   268 ; MODULE FOR$CVTRT - replaces FOR$CNV_OUT
01A8   269          OLDENTRY        FOR$CNV_OUT_D
01A8   270          MAC     CALL    FOR     FOR$CVT_D_TD
01B0   271          OLDENTRY        FOR$CNV_OUT_E
01B0   272          MAC     CALL    FOR     FOR$CVT_D_TE
01B8   273          OLDENTRY        FOR$CNV_OUT_F
01B8   274          MAC     CALL    FOR     FOR$CVT_D_TF
01C0   275          OLDENTRY        FOR$CNV_OUT_G
01C0   276          MAC     CALL    FOR     FOR$CVT_D_TG
01C8   277
01C8   278 ; MODULE:FOR$DATE
01C8   279          MAC     NOVECT  FOR     FOR$DATE
```

```
01C8    280
01C8    281 ; MODULE:FOR$DATE_T_DS
01C8    282         MAC     NOVECT  FOR     FOR$DATE_T_DS
01C8    283
01C8    284 ; MODULE:FOR$DEFINE_FILE
01C8    285         MAC     CALL    FOR     FOR$DEF_FILE
01D0    286         MAC     CALL    FOR     FOR$DEF_FILE_W
01D8    287
01D8    288 ; MOUDLE FOR$ENDFILE
01D8    289         MAC     CALL    FOR     FOR$ENDFILE
01E0    290
01E0    291 ; MODULE:FOR$ENODEF
01E0    292         MAC     SYM     FOR     FOR$K_ADJARRDIM
01E0    293         MAC     SYM     FOR     FOR$K_ARRREFOUT
01E0    294         MAC     SYM     FOR     FOR$K_ATTACCNON
01E0    295         MAC     SYM     FOR     FOR$K_BACERR
01E0    296
01E0    297         MAC     SYM     FOR     FOR$K_CLOERR
01E0    298         MAC     SYM     FOR     FOR$K_DECSTROVE
01E0    299         MAC     SYM     FOR     FOR$K_DELERR
01E0    300         MAC     SYM     FOR     FOR$K_DUPFILSPE
01E0    301         MAC     SYM     FOR     FOR$K_ENDDURREA
01E0    302         MAC     SYM     FOR     FOR$K_ENDFILERR
01E0    303         MAC     SYM     FOR     FOR$K_ERRDURREA
01E0    304         MAC     SYM     FOR     FOR$K_ERRDURWRI
01E0    305         MAC     SYM     FOR     FOR$K_FAC_NO
01E0    306
01E0    307         MAC     SYM     FOR     FOR$K_FILNAMSPE
01E0    308         MAC     SYM     FOR     FOR$K_FILNOTFOU
01E0    309         MAC     SYM     FOR     FOR$K_FINERR
01E0    310         MAC     SYM     FOR     FOR$K_FLOOVE
01E0    311         MAC     SYM     FOR     FOR$K_FLOUND
01E0    312
01E0    313         MAC     SYM     FOR     FOR$K_FLOZERDIV
01E0    314         MAC     SYM     FOR     FOR$K_FORVARMIS
01E0    315         MAC     SYM     FOR     FOR$K_INCFILORG
01E0    316         MAC     SYM     FOR     FOR$K_INCKEYCHG
01E0    317         MAC     SYM     FOR     FOR$K_INCOPECLO
01E0    318         MAC     SYM     FOR     FOR$K_INCRECLEN
01E0    319         MAC     SYM     FOR     FOR$K_INCRECTYP
01E0    320         MAC     SYM     FOR     FOR$K_INFFORLOO
01E0    321         MAC     SYM     FOR     FOR$K_INPCONERR
01E0    322         MAC     SYM     FOR     FOR$K_INPRECTOO
01E0    323         MAC     SYM     FOR     FOR$K_INPSTAREQ
01E0    324         MAC     SYM     FOR     FOR$K_INSVIRMEM
01E0    325         MAC     SYM     FOR     FOR$K_INTOVF
01E0    326         MAC     SYM     FOR     FOR$K_INTZERDIV
01E0    327         MAC     SYM     FOR     FOR$K_INVARGFOR
01E0    328         MAC     SYM     FOR     FOR$K_INVKEYSPE
01E0    329         MAC     SYM     FOR     FOR$K_INVLOGUNI
01E0    330         MAC     SYM     FOR     FOR$K_INVREFVAR
01E0    331         MAC     SYM     FOR     FOR$K_KEYVALERR
01E0    332         MAC     SYM     FOR     FOR$K_LISIO_SYN
01E0    333         MAC     SYM     FOR     FOR$K_MAX_ERR
01E0    334         MAC     SYM     FOR     FOR$K_MIXFILACC
01E0    335         MAC     SYM     FOR     FOR$K_NOTFORSPE
01E0    336         MAC     SYM     FOR     FOR$K_NO_CURREC
```

```
01E0   337          MAC     SYM     FOR     FOR$K_NO_SUCDEV
01E0   338          MAC     SYM     FOR     FOR$K_OPEDEFREQ
01E0   339          MAC     SYM     FOR     FOR$K_OPEFAI
01E0   340
01E0   341          MAC     SYM     FOR     FOR$K_OUTCONERR
01E0   342          MAC     SYM     FOR     FOR$K_OUTSTAOVE
01E0   343          MAC     SYM     FOR     FOR$K_RECIO_OPE
01E0   344          MAC     SYM     FOR     FOR$K_RECNUMOUT
01E0   345          MAC     SYM     FOR     FOR$K_REWERR
01E0   346          MAC     SYM     FOR     FOR$K_REWRITERR
01E0   347          MAC     SYM     FOR     FOR$K_SEGRECFOR
01E0   348          MAC     SYM     FOR     FOR$K_SPERECLOC
01E0   349          MAC     SYM     FOR     FOR$K_SYNERRFOR
01E0   350          MAC     SYM     FOR     FOR$K_SYNERRNAM
01E0   351          MAC     SYM     FOR     FOR$K_TOOMANREC
01E0   352          MAC     SYM     FOR     FOR$K_TOOMANVAL
01E0   353          MAC     SYM     FOR     FOR$K_UNIALROPE
01E0   354          MAC     SYM     FOR     FOR$K_UNLERR
01E0   355          MAC     SYM     FOR     FOR$K_VFEVALERR
01E0   356          MAC     SYM     FOR     FOR$K_WRIREAFIL
01E0   357
01E0   358 ; MODULE:FOR$ERRSNS                  ; See also end where FOR$$ERRSNS_SAV declared
01E0   359          MAC     CALL    FOR     FOR$ERRSNS
01E8   360          MAC     CALL    FOR     FOR$ERRSNS_W
01F0   361
01F0   362 ; MODULE:FOR$EXIT
01F0   363          MAC     CALL    FOR     FOR$EXIT
01F8   364          MAC     CALL    FOR     FOR$EXIT_W
0200   365
0200   366 ; MODULE:OTS$CVTTR
0200   367                                  ; This module is a replacement
0200   368                                  ; for FOR$CNVIR.  The old
0200   369                                  ; FOR$ entry point still exists.
0200   370                                  ; See later where other OTS$
0200   371                                  ; entry points are named.
0200   372
0200   373          OLDENTRY        FOR$CNV_IN_DEFG ; Same as next symbol.
0200   374          MAC     CALL    OTS     OTS$CVT_T_D
0208   375
0208   376 ; MODULE:FOR$FIND
0208   377          MAC     CALL    FOR     FOR$FIND
0210   378
0210   379 ; MODULE:OTS$CVT_TI_L - replaces in part FOR$CNVII
0210   380          OLDENTRY        FOR$CNV_IN_I
0210   381          MAC     CALL    OTS     OTS$CVT_TI_L
0218   382
0218   383 ; MODULE:OTS$CVT_TL_L - replaces in part FOR$CNVII
0218   384          OLDENTRY        FOR$CNV_IN_L
0218   385          MAC     CALL    OTS     OTS$CVT_TL_L
0220   386
0220   387 ; MODULE: OTS$CVT_TO_L - replaces in part FOR$CNVII
0220   388          OLDENTRY        FOR$CNV_IN_O
0220   389          MAC     CALL    OTS     OTS$CVT_TO_L
0228   390          OLDENTRY        FOR$CNV_IN_Z
0228   391          MAC     CALL    OTS     OTS$CVT_TZ_L
0230   392
0230   393 ; MODULE:FOR$IDATE
```

```
0230   394          MAC     NOVECT  FOR     FOR$IDATE
0230   395
0230   396 ; MODULE:FOR$INI_DES
0230   397          MAC     JSB     FOR     FOR$INI_DES1_R2
0238   398          MAC     JSB     FOR     FOR$INI_DES2_R3
0240   399          MAC     JSB     FOR     FOR$INI_DESC_R6
0248   400
0248   401 ; MODULE:FOR$JDATE
0248   402          MAC     NOVECT  FOR     FOR$JDATE
0248   403
0248   404 ; MODULE:FOR$MSGDEF
0248   405          MAC     SYM     FOR     FOR$_ADJARRDIM
0248   406          MAC     SYM     FOR     FOR$_ATTACCNON
0248   407          MAC     SYM     FOR     FOR$_BACERR
0248   408          MAC     SYM     FOR     FOR$_CLOERR
0248   409
0248   410          MAC     SYM     FOR     FOR$_DELERR
0248   411          MAC     SYM     FOR     FOR$_DUPFILSPE
0248   412          MAC     SYM     FOR     FOR$_ENDDURREA
0248   413          MAC     SYM     FOR     FOR$_ENDFILERR
0248   414          MAC     SYM     FOR     FOR$_ERRDURREA
0248   415
0248   416          MAC     SYM     FOR     FOR$_ERRDURWRI
0248   417          MAC     SYM     FOR     FOR$_FILNAMSPE
0248   418          MAC     SYM     FOR     FOR$_FILNOTFOU
0248   419          MAC     SYM     FOR     FOR$_FINERR
0248   420          MAC     SYM     FOR     FOR$_FORVARMIS
0248   421
0248   422          MAC     SYM     FOR     FOR$_INCFILORG
0248   423          MAC     SYM     FOR     FOR$_INCKEYCHG
0248   424          MAC     SYM     FOR     FOR$_INCOPECLO
0248   425          MAC     SYM     FOR     FOR$_INCRECLEN
0248   426          MAC     SYM     FOR     FOR$_INCRECTYP
0248   427          MAC     SYM     FOR     FOR$_INFFORLOO
0248   428          MAC     SYM     FOR     FOR$_INPCONERR
0248   429
0248   430          MAC     SYM     FOR     FOR$_INPRECTOO
0248   431          MAC     SYM     FOR     FOR$_INPSTAREQ
0248   432          MAC     SYM     FOR     FOR$_INSVIRMEM
0248   433          MAC     SYM     FOR     FOR$_INVARGFOR
0248   434          MAC     SYM     FOR     FOR$_INVKEYSPE
0248   435
0248   436          MAC     SYM     FOR     FOR$_INVLOGUNI
0248   437          MAC     SYM     FOR     FOR$_INVREFVAR
0248   438          MAC     SYM     FOR     FOR$_KEYVALERR
0248   439          MAC     SYM     FOR     FOR$_LISIO_SYN
0248   440          MAC     SYM     FOR     FOR$_MIXFILACC
0248   441
0248   442          MAC     SYM     FOR     FOR$_NOTFORSPE
0248   443          MAC     SYM     FOR     FOR$_NO_CURREC
0248   444          MAC     SYM     FOR     FOR$_NO_SUCDEV
0248   445          MAC     SYM     FOR     FOR$_OPEDEFREQ
0248   446
0248   447          MAC     SYM     FOR     FOR$_OPEFAI
0248   448          MAC     SYM     FOR     FOR$_OUTCONERR
0248   449          MAC     SYM     FOR     FOR$_OUTSTAOVE
0248   450          MAC     SYM     FOR     FOR$_RECIO_OPE
```

```
0248   451
0248   452          MAC     SYM     FOR     FOR$_RECNUMOUT
0248   453          MAC     SYM     FOR     FOR$_REWERR
0248   454          MAC     SYM     FOR     FOR$_REWPITERR
0248   455          MAC     SYM     FOR     FOR$_SEGRECFOR
0248   456
0248   457          MAC     SYM     FOR     FOR$_SPERECLOC
0248   458          MAC     SYM     FOR     FOR$_SYNERRFOR
0248   459          MAC     SYM     FOR     FOR$_SYNERRNAM
0248   460          MAC     SYM     FOR     FOR$_TOOMANREC
0248   461          MAC     SYM     FOR     FOR$_TOOMANVAL
0248   462          MAC     SYM     FOR     FOR$_UNIALROPE
0248   463          MAC     SYM     FOR     FOR$_UNLERR
0248   464          MAC     SYM     FOR     FOR$_VFEVALERR
0248   465          MAC     SYM     FOR     FOR$_WRIREAFIL
0248   466
0248   467 ; MODULE:FOR$PAUSE
0248   468          MAC     CALL    FOR     FOR$PAUSE
0250   469
0250   470 ; MODULE:FOR$RANDOM
0250   471          MAC     NOVECT  FOR     FOR$IRAN
0250   472          MAC     NOVECT  FOR     FOR$RANDU
0250   473          MAC     NOVECT  FOR     FOR$RANDU_W
0250   474
0250   475 ; MODULE:FOR$REWIND
0250   476          MAC     CALL    FOR     FOR$REWIND
0258   477
0258   478 ; MODULE:FOR$SECNDS
0258   479          MAC     CALL    FOR     FOR$SECNDS
0260   480
0260   481 ; MODULE:FOR$STOP
0260   482          MAC     CALL    FOR     FOR$STOP
0268   483
0268   484 ; MODULE:FOR$TIME
0268   485          MAC     NOVECT  FOR     FOR$TIME
0268   486
0268   487 ; MODULE:FOR$TIME_T_DS
0268   488          MAC     NOVECT  FOR     FOR$TIME_T_DS
0268   489
```

```
0268    491 ;+
0268    492 ; Mathematical library entry points
0268    493 ; Include the frequently used ones first (ones with JSBs)
0268    494 ;-
0268    495
0268    496 ; MODULE:MTH$ACOS (Degree entries further on)
0268    497         MAC     CALL    MTH     MTH$ACOS
0270    498         OLDENTRY        MTH$ACOS_R5             ; Release 1 name
0270    499         MAC     JSB     MTH     MTH$ACOS_R4
0278    500
0278    501 ; MODULE:MTH$ALOG
0278    502         MAC     CALL    MTH     MTH$ALOG
0280    503         MAC     CALL    MTH     MTH$ALOG10
0288    504         MAC     JSB     MTH     MTH$ALOG10_R5
0290    505         MAC     JSB     MTH     MTH$ALOG_R5
0298    506
0298    507 ; MODULE:MTH$ASIN
0298    508         MAC     CALL    MTH     MTH$ASIN
02A0    509         OLDENTRY        MTH$ASIN_R5             ; Release 1 name
02A0    510         MAC     JSB     MTH     MTH$ASIN_R4
02A8    511
02A8    512 ; MODULE:MTH$ATAN
02A8    513         MAC     CALL    MTH     MTH$ATAN
02B0    514         MAC     CALL    MTH     MTH$ATAN2
02B8    515         MAC     JSB     MTH     MTH$ATAN_R4
02C0    516
02C0    517 ; MODULE:MTH$DACOS
02C0    518         MAC     CALL    MTH     MTH$DACOS
02C8    519         OLDENTRY        MTH$DACOS_R9            ; Release 1 name
02C8    520         MAC     JSB     MTH     MTH$DACOS_R7
02D0    521
02D0    522 ; MODULE:MTH$DASIN
02D0    523         MAC     CALL    MTH     MTH$DASIN
02D8    524         OLDENTRY        MTH$DASIN_R9            ; Release 1 name
02D8    525         MAC     JSB     MTH     MTH$DASIN_R7
02E0    526
02E0    527 ; MODULE:MTH$DATAN
02E0    528         MAC     CALL    MTH     MTH$DATAN
02E8    529         MAC     CALL    MTH     MTH$DATAN2
02F0    530         MAC     JSB     MTH     MTH$DATAN_R7
02F8    531
02F8    532 ; MODULE:MTH$DEXP
02F8    533         MAC     CALL    MTH     MTH$DEXP
0300    534         OLDENTRY        MTH$DEXP_R7             ; Obsolete name
0300    535         MAC     JSB     MTH     MTH$DEXP_R6
```

```
0308    537 ; MODULE:MTH$DLOG
0308    538         MAC     CALL    MTH     MTH$DLOG
0310    539         MAC     CALL    MTH     MTH$DLOG10
0318    540         MAC     JSB     MTH     MTH$DLOG10_R8
0320    541         MAC     JSB     MTH     MTH$DLOG_R8
0328    542
0328    543 ; MODULE:MTH$DSINCOS
0328    544         MAC     CALL    MTH     MTH$DCOS
0330    545         MAC     JSB     MTH     MTH$DCOS_R7
0338    546         MAC     CALL    MTH     MTH$DSIN
0340    547         MAC     JSB     MTH     MTH$DSIN_R7
0348    548
0348    549 ; MODULE:MTH$DSQRT
0348    550         MAC     CALL    MTH     MTH$DSQRT
0350    551         MAC     JSB     MTH     MTH$DSQRT_R5
0358    552
0358    553 ; MODULE:MTH$EXP
0358    554         MAC     CALL    MTH     MTH$EXP
0360    555         MAC     JSB     MTH     MTH$EXP_R4
0368    556
0368    557 ; MODULE:MTH$SINCOS
0368    558         MAC     CALL    MTH     MTH$COS
0370    559         MAC     JSB     MTH     MTH$COS_R4
0378    560         MAC     CALL    MTH     MTH$SIN
0380    561         MAC     JSB     MTH     MTH$SIN_R4
0388    562
0388    563 ; MODULE:MTH$SQRT
0388    564         MAC     CALL    MTH     MTH$SQRT
0390    565 ;       JSB to MTH$SQRT_R3 is with new entries.
0390    566
0390    567 ; MODULE:MTH$SQRTR2     (obsolete module)
0390    568         MAC     JSB     MTH     MTH$SQRT_R2
0398    569
```

```
0398    571 ;+
0398    572 ; Language independent support entry points
0398    573 ; Include them after frequently used math routines, since
0398    574 ; they have the power routines.
0398    575 ;-
0398    576
0398    577
0398    578 ; MODULE:OTS$DIVC
0398    579         MAC     CALL    OTS     OTS$DIVC
03A0    580
03A0    581 ; MODULE:OTS$LINKAGE
03A0    582         MAC     SYM     OTS     OTS$LINKAGE
03A0    583
03A0    584 ; MODULE:OTS$MSGDEF
03A0    585         MAC     SYM     OTS     OTS$_FATINTERR
03A0    586         MAC     SYM     OTS     OTS$_INPCONERR
03A0    587         MAC     SYM     OTS     OTS$_INTDATCOR
03A0    588         MAC     SYM     OTS     OTS$_INVSTRDES
03A0    589         MAC     SYM     OTS     OTS$_IO_CONCLO
03A0    590         MAC     SYM     OTS     OTS$_OUTCONERR
03A0    591         MAC     SYM     OTS     OTS$_USEFLORES
03A0    592         MAC     SYM     OTS     OTS$_WRONUMARG
03A0    593
03A0    594 ; MODULE:OTS$POWCJ
03A0    595         MAC     CALL    OTS     OTS$POWCJ
03A8    596
03A8    597 ; MODULE:OTS$POWDD
03A8    598         MAC     CALL    OTS     OTS$POWDD
03B0    599         MAC     CALL    OTS     OTS$POWDR
03B8    600         MAC     CALL    OTS     OTS$POWRD
03C0    601
03C0    602 ; MODULE:OTS$POWDJ
03C0    603         MAC     CALL    OTS     OTS$POWDJ
03C8    604
03C8    605 ; MODULE:OTS$POWII
03C8    606         MAC     CALL    OTS     OTS$POWII
03D0    607
03D0    608 ; MODULE:OTS$POWJJ
03D0    609         MAC     CALL    OTS     OTS$POWJJ
03D8    610
03D8    611 ; MODULE:OTS$POWRJ
03D8    612         MAC     CALL    OTS     OTS$POWRJ
03E0    613
03E0    614 ; MODULE:OTS$POWRR
03E0    615         MAC     CALL    OTS     OTS$POWRR
03E8    616
03E8    617 ; MODULE:OTS$SCOPY
03E8    618         MAC     CALL    OTS     OTS$SCOPY_DXDX
03F0    619         MAC     JSB     OTS     OTS$SCOPY_DXDX6
03F8    620         MAC     CALL    OTS     OTS$SCOPY_R_DX
0400    621         MAC     JSB     OTS     OTS$SCOPY_R_DX6
0408    622         MAC     CALL    OTS     OTS$SGET1_DD
0410    623         MAC     JSB     OTS     OTS$SGET1_DD_R6
0418    624         MAC     CALL    OTS     OTS$SFREE1_DD
0420    625         MAC     JSB     OTS     OTS$SFREE1_DD6
0428    626         MAC     CALL    OTS     OTS$SFREEN_DD
0430    627         MAC     JSB     OTS     OTS$SFREEN_DD6
```

```
0438    629 ;+
0438    630 ; Now define the rest of the Math entry points
0438    631 ;-
0438    632
0438    633 ; MODULE:MTH$ABS
0438    634         MAC     NOVECT  MTH     MTH$ABS
0438    635         MAC     NOVECT  MTH     MTH$DABS
0438    636         MAC     NOVECT  MTH     MTH$GABS
0438    637         MAC     NOVECT  MTH     MTH$HABS
0438    638         MAC     NOVECT  MTH     MTH$IIABS
0438    639         MAC     NOVECT  MTH     MTH$JIABS
0438    640
0438    641 ; MODULE:MTH$AINT
0438    642         MAC     NOVECT  MTH     MTH$AINT
0438    643
0438    644 ; MODULE:MTH$AMOD
0438    645         MAC     NOVECT  MTH     MTH$AMOD
0438    646
0438    647 ; MODULE:MTH$ANINT
0438    648         MAC     NOVECT  MTH     MTH$ANINT
0438    649
0438    650 ; MODULE:MTH$BITOPS
0438    651         MAC     NOVECT  MTH     MTH$IIAND
0438    652         MAC     NOVECT  MTH     MTH$IIEOR
0438    653         MAC     NOVECT  MTH     MTH$IIOR
0438    654         MAC     NOVECT  MTH     MTH$IISHFT
0438    655
0438    656         MAC     NOVECT  MTH     MTH$INOT
0438    657         MAC     NOVECT  MTH     MTH$JIAND
0438    658         MAC     NOVECT  MTH     MTH$JIEOR
0438    659         MAC     NOVECT  MTH     MTH$JIOR
0438    660
0438    661         MAC     NOVECT  MTH     MTH$JISHFT
0438    662         MAC     NOVECT  MTH     MTH$JNOT
0438    663
0438    664 ; MODULE:MTH$CABS
0438    665         MAC     CALL    MTH     MTH$CABS
0440    666
0440    667 ; MODULE:MTH$CEXP
0440    668         MAC     CALL    MTH     MTH$CEXP
0448    669
0448    670 ; MODULE:MTH$CLOG
0448    671         MAC     CALL    MTH     MTH$CLOG
0450    672
0450    673 ; MODULE:MTH$CONJG
0450    674         MAC     NOVECT  MTH     MTH$CONJG
0450    675
0450    676 ; MODULE:MTH$CONVER
0450    677         MAC     NOVECT  MTH     MTH$AIMAG
0450    678         MAC     NOVECT  MTH     MTH$DIMAG
0450    679         MAC     NOVECT  MTH     MTH$GIMAG
0450    680         MAC     NOVECT  MTH     MTH$CMPLX
0450    681         MAC     NOVECT  MTH     MTH$DCMPLX
0450    682         MAC     NOVECT  MTH     MTH$GCMPLX
0450    683         MAC     NOVECT  MTH     MTH$DBLE
0450    684         MAC     NOVECT  MTH     MTH$GDBLE
0450    685         MAC     NOVECT  MTH     MTH$DFLOTI
```

```
0450   686          MAC     NOVECT  MTH     MTH$DFLOTJ
0450   687          MAC     NOVECT  MTH     MTH$FLOATI
0450   688          MAC     NOVECT  MTH     MTH$FLOATJ
0450   689          MAC     NOVECT  MTH     MTH$GFLOTI
0450   690          MAC     NOVECT  MTH     MTH$GFLOTJ
0450   691          MAC     NOVECT  MTH     MTH$IIDINT
0450   692          MAC     NOVECT  MTH     MTH$IIGINT
0450   693          MAC     NOVECT  MTH     MTH$IIHINT
0450   694          MAC     NOVECT  MTH     MTH$IIFIX
0450   695          MAC     NOVECT  MTH     MTH$IINT
0450   696          MAC     NOVECT  MTH     MTH$JIDINT
0450   697          MAC     NOVECT  MTH     MTH$JIGINT
0450   698          MAC     NOVECT  MTH     MTH$JIHINT
0450   699          MAC     NOVECT  MTH     MTH$JIFIX
0450   700          MAC     NOVECT  MTH     MTH$JINT
0450   701          MAC     NOVECT  MTH     MTH$REAL
0450   702          MAC     NOVECT  MTH     MTH$DREAL
0450   703          MAC     NOVECT  MTH     MTH$GREAL
0450   704          MAC     NOVECT  MTH     MTH$SNGL
0450   705          MAC     NOVECT  MTH     MTH$SNGLG
0450   706
0450   707 ; MODULE:MTH$COSH
0450   708          MAC     CALL    MTH     MTH$COSH
0458   709
0458   710 ; MODULE:MTH$CSINCOS
0458   711          MAC     CALL    MTH     MTH$CCOS
0460   712          MAC     CALL    MTH     MTH$CSIN
0468   713
0468   714 ; MODULE:MTH$CSQRT
0468   715          MAC     CALL    MTH     MTH$CSQRT
0470   716
0470   717 ; MODULE:MTH$DCOSH
0470   718          MAC     CALL    MTH     MTH$DCOSH
0478   719
0478   720 ; MODULE:MTH$DIM
0478   721          MAC     NOVECT  MTH     MTH$DDIM
0478   722          MAC     NOVECT  MTH     MTH$DIM
0478   723          MAC     NOVECT  MTH     MTH$IIDIM
0478   724          MAC     NOVECT  MTH     MTH$JIDIM
0478   725
0478   726 ; MODULE:MTH$DINT
0478   727          MAC     NOVECT  MTH     MTH$DINT
```

```
0478    729
0478    730 ; MODULE:MTH$DMAX1
0478    731         MAC     NOVECT  MTH     MTH$DMAX1
0478    732
0478    733 ; MODULE:MTH$DMIN1
0478    734         MAC     NOVECT  MTH     MTH$DMIN1
0478    735
0478    736 ; MODULE:MTH$DMOD
0478    737         MAC     NOVECT  MTH     MTH$DMOD
0478    738
0478    739 ; MODULE:MTH$DNINT
0478    740         MAC     NOVECT  MTH     MTH$DNINT
0478    741
0478    742 ; MODULE:MTH$DPROD
0478    743         MAC     NOVECT  MTH     MTH$DPROD
0478    744
0478    745 ; MODULE:MTH$DSIGN
0478    746         MAC     NOVECT  MTH     MTH$DSIGN
0478    747
0478    748 ; MODULE:MTH$DSINH
0478    749         MAC     CALL    MTH     MTH$DSINH
0480    750
0480    751 ; MODULE:MTH$DTAN
0480    752         MAC     CALL    MTH     MTH$DTAN
0488    753
0488    754 ; MODULE:MTH$DTANH
0488    755         MAC     CALL    MTH     MTH$DTANH
0490    756
0490    757 ; MODULE:MTH$IIDNNT
0490    758         MAC     NOVECT  MTH     MTH$IIDNNT
0490    759
0490    760 ; MODULE:MTH$IISIGN
0490    761         MAC     NOVECT  MTH     MTH$IISIGN
0490    762
0490    763 ; MODULE:MTH$IMAX0
0490    764         MAC     NOVECT  MTH     MTH$AIMAX0
0490    765         MAC     NOVECT  MTH     MTH$IMAX0
0490    766
0490    767 ; MODULE:MTH$IMIN0
0490    768         MAC     NOVECT  MTH     MTH$AIMIN0
0490    769         MAC     NOVECT  MTH     MTH$IMIN0
0490    770
0490    771 ; MODULE:MTH$ININT
0490    772         MAC     NOVECT  MTH     MTH$ININT
0490    773
0490    774 ; MODULE:MTH$JIDNNT
0490    775         MAC     NOVECT  MTH     MTH$JIDNNT
0490    776
0490    777 ; MODULE:MTH$JISIGN
0490    778         MAC     NOVECT  MTH     MTH$JISIGN
0490    779
0490    780 ; MODULE:MTH$JMAX0
0490    781         MAC     NOVECT  MTH     MTH$AJMAX0
0490    782         MAC     NOVECT  MTH     MTH$JMAX0
0490    783
0490    784 ; MODULE:MTH$JMIN0
0490    785         MAC     NOVECT  MTH     MTH$AJMIN0
```

```
0490   786          MAC     NOVECT  MTH     MTH$JMINO
0490   787
0490   788 ; MODULE:MTH$JNINT
0490   789          MAC     NOVECT  MTH     MTH$JNINT
0490   790
0490   791 ; MODULE:MTH$MAX1
0490   792          MAC     NOVECT  MTH     MTH$AMAX1
0490   793          MAC     NOVECT  MTH     MTH$IMAX1
0490   794          MAC     NOVECT  MTH     MTH$JMAX1
0490   795
0490   796 ; MODULE:MTH$MIN1
0490   797          MAC     NOVECT  MTH     MTH$AMIN1
0490   798          MAC     NOVECT  MTH     MTH$IMIN1
0490   799          MAC     NOVECT  MTH     MTH$JMIN1
0490   800
0490   801 ; MODULE:MTH$MOD
0490   802          MAC     NOVECT  MTH     MTH$IMOD
0490   803          MAC     NOVECT  MTH     MTH$JMOD
0490   804
0490   805 ; MODULE:MTH$MSGDEF
0490   806          MAC     SYM     MTH     MTH$_FLOOVEMAT
0490   807          MAC     SYM     MTH     MTH$_FLOUNDMAT
0490   808          MAC     SYM     MTH     MTH$_INVARGMAT
0490   809          MAC     SYM     MTH     MTH$_LOGZERNEG
0490   810
0490   811          MAC     SYM     MTH     MTH$_SIGLOSMAT
0490   812
0490   813          MAC     SYM     MTH     MTH$_SQUROONEG
0490   814          MAC     SYM     MTH     MTH$_UNDEXP
0490   815          MAC     SYM     MTH     MTH$_WRONUMARG
0490   816
0490   817 ; MODULE:MTH$RANDOM
0490   818          MAC     CALL    MTH     MTH$RANDOM
0498   819
0498   820 ; MODULE:MTH$SIGN
0498   821          MAC     NOVECT  MTH     MTH$SIGN
0498   822
0498   823 ; MODULE:MTH$SINH
0498   824          MAC     CALL    MTH     MTH$SINH
04A0   825
04A0   826 ; MODULE:MTH$TAN
04A0   827          MAC     CALL    MTH     MTH$TAN
04A8   828
04A8   829 ; MODULE:MTH$TANH
04A8   830          MAC     CALL    MTH     MTH$TANH
04B0   831
```

```
04B0    833 ;+
04B0    834 ; General library entry points LIB$
04B0    835 ;-
04B0    836
04B0    837
04B0    838 ; MODULE:LIB$AST_IN_PROG
04B0    839         MAC     CALL    LIB     LIB$AST_IN_PROG
04B8    840
04B8    841 ; MODULE:LIB$CHAR
04B8    842         MAC     NOVECT  LIB     LIB$CHAR
04B8    843
04B8    844 ; MODULE:LIB$CRC
04B8    845         MAC     CALL    LIB     LIB$CRC
04C0    846
04C0    847 ; MODULE:LIB$CRC_TABLE
04C0    848         MAC     CALL    LIB     LIB$CRC_TABLE
04C8    849
04C8    850 ; MODULE:LIB$CVTDF
04C8    851         MAC     NOVECT  LIB     LIB$CVTDF
04C8    852
04C8    853
04C8    854 ; MODULE:LIB$DEC_OVER
04C8    855         MAC     CALL    LIB     LIB$DEC_OVER
04D0    856
04D0    857 ; MODULE:LIB$ESTABLISH
04D0    858         MAC     CALL    LIB     LIB$ESTABLISH
04D8    859
04D8    860 ; MODULE:LIB$EXTV
04D8    861         MAC     CALL    LIB     LIB$EXTV
04E0    862
04E0    863 ; MODULE:LIB$EXTZV
04E0    864         MAC     CALL    LIB     LIB$EXTZV
04E8    865
04E8    866 ; MODULE:LIB$FFC
04E8    867         MAC     CALL    LIB     LIB$FFC
04F0    868
04F0    869 ; MODULE:LIB$FFS
04F0    870         MAC     CALL    LIB     LIB$FFS
04F8    871
04F8    872 ; MODULE:LIB$FIXUP_FLT
04F8    873         MAC     CALL    LIB     LIB$FIXUP_FLT
0500    874
0500    875 ; MODULE:LIB$FLT_UNDER
0500    876         MAC     CALL    LIB     LIB$FLT_UNDER
0508    877
0508    878 ; MODULE:LIB$GET_INPUT
0508    879         MAC     CALL    LIB     LIB$GET_INPUT
0510    880         MAC     CALL    LIB     LIB$GET_COMMAND
0518    881
0518    882 ; MODULE:LIB$ICHAR
0518    883         MAC     NOVECT  LIB     LIB$ICHAR
0518    884
0518    885 ; MODULE:LIB$INDEX
0518    886         MAC     CALL    LIB     LIB$INDEX
0520    887
0520    888 ; MODULE:LIB$INITIALIZE
0520    889         MAC     NOVECT  LIB     LIB$INITIALIZE
```

```
0520    890
0520    891 ; MODULE:LIB$INSV
0520    892         MAC     CALL    LIB     LIB$INSV
0528    893
0528    894 ; MODULE:LIB$INT_OVER
0528    895         MAC     CALL    LIB     LIB$INT_OVER
0530    896
0530    897 ; MODULE:LIB$LEN
0530    898         MAC     NOVECT  LIB     LIB$LEN
0530    899
0530    900 ; MODULE:LIB$LOCC
0530    901         MAC     CALL    LIB     LIB$LOCC
0538    902
0538    903 ; MODULE:LIB$LOOKUP_KEY
0538    904         MAC     NOVECT  LIB     LIB$LOOKUP_KEY
0538    905
0538    906 ; MODULE:LIB$MATCHC
0538    907         MAC     CALL    LIB     LIB$MATCHC
0540    908
0540    909 ; MODULE:LIB$MATCH_COND
0540    910         MAC     CALL    LIB     LIB$MATCH_COND
0548    911
0548    912 ; MODULE:LIB$MOVTC
0548    913         MAC     CALL    LIB     LIB$MOVTC
0550    914
0550    915 ; MODULE:LIB$MOVTUC
0550    916         MAC     CALL    LIB     LIB$MOVTUC
0558    917
0558    918 ; MODULE:LIB$MSGDEF
0558    919         MAC     SYM     LIB     LIB$_AMBKEY
0558    920         MAC     SYM     LIB     LIB$_ATTCONSTO
0558    921         MAC     SYM     LIB     LIB$_BADBLOADR
0558    922         MAC     SYM     LIB     LIB$_BADBLOSIZ
0558    923         MAC     SYM     LIB     LIB$_BADSTA
0558    924         MAC     SYM     LIB     LIB$_EF_ALRFRE
0558    925         MAC     SYM     LIB     LIB$_EF_ALRRES
0558    926         MAC     SYM     LIB     LIB$_EF_RESSYS
0558    927         MAC     SYM     LIB     LIB$_FATERRLIB
0558    928         MAC     SYM     LIB     LIB$_INPSTRTRU
0558    929         MAC     SYM     LIB     LIB$_INSEF
0558    930         MAC     SYM     LIB     LIB$_INSVIRMEM
0558    931
0558    932         MAC     SYM     LIB     LIB$_INTLOGERR
0558    933         MAC     SYM     LIB     LIB$_INVARG
0558    934         MAC     SYM     LIB     LIB$_INVSTRDES
0558    935         MAC     SYM     LIB     LIB$_NORMAL
0558    936
0558    937         MAC     SYM     LIB     LIB$_NOTFOU
0558    938         MAC     SYM     LIB     LIB$_PUSSTAOVE
0558    939         MAC     SYM     LIB     LIB$_SIGNO_ARG
0558    940         MAC     SYM     LIB     LIB$_STRIS_INT
0558    941         MAC     SYM     LIB     LIB$_STRTRU
0558    942
0558    943         MAC     SYM     LIB     LIB$_UNRKEY
0558    944         MAC     SYM     LIB     LIB$_USEFLORES
0558    945         MAC     SYM     LIB     LIB$_WRONUMARG
0558    946
```

```
0558   947 ; MODULE:LIB$PUT_OUTPUT
0558   948         MAC     CALL    LIB     LIB$PUT_OUTPUT
0560   949
0560   950 ; MODULE:LIB$REVERT
0560   951         MAC     CALL    LIB     LIB$REVERT
0568   952
0568   953 ; MODULE:LIB$SCANC
0568   954         MAC     CALL    LIB     LIB$SCANC
0570   955
0570   956 ; MODULE:LIB$SCOPY
0570   957         MAC     CALL    LIB     LIB$SCOPY_DXDX
0578   958         MAC     JSB     LIB     LIB$SCOPY_DXDX6
0580   959         MAC     CALL    LIB     LIB$SCOPY_R_DX
0588   960         MAC     JSB     LIB     LIB$SCOPY_R_DX6
0590   961         MAC     CALL    LIB     LIB$SGET1_DD
0598   962         MAC     JSB     LIB     LIB$SGET1_DD_R6
05A0   963         MAC     CALL    LIB     LIB$SFREE1_DD
05A8   964         MAC     JSB     LIB     LIB$SFREE1_DD6
05B0   965         MAC     CALL    LIB     LIB$SFREEN_DD
05B8   966         MAC     JSB     LIB     LIB$SFREEN_DD6
05C0   967
05C0   968 ; MODULE:LIB$STAT_VM
05C0   969         MAC     CALL    LIB     LIB$STAT_VM
05C8   970
05C8   971 ; MODULE:LIB$SIGNAL
05C8   972         MAC     CALL    LIB     LIB$SIGNAL
05D0   973         MAC     CALL    LIB     LIB$STOP
05D8   974
05D8   975 ; MODULE:LIB$SIG_TO_RET
05D8   976
05D8   977         MAC     CALL    LIB     LIB$SIG_TO_RET
05E0   978
05E0   979 ; MODULE:LIB$SKPC
05E0   980         MAC     CALL    LIB     LIB$SKPC
05E8   981
05E8   982 ; MODULE:LIB$SPANC
05E8   983         MAC     CALL    LIB     LIB$SPANC
05F0   984
05F0   985 ; MODULE:LIB$SYS_ASCTIM
05F0   986         MAC     NOVECT  LIB     LIB$SYS_ASCTIM
05F0   987
05F0   988 ; MODULE:LIB$SYS_FAO
05F0   989         MAC     NOVECT  LIB     LIB$SYS_FAO
05F0   990
05F0   991 ; MODULE:LIB$SYS_FAOL
05F0   992         MAC     NOVECT  LIB     LIB$SYS_FAOL
05F0   993
05F0   994 ; MODULE:LIB$SYS_GETMSG
05F0   995         MAC     NOVECT  LIB     LIB$SYS_GETMSG
05F0   996
05F0   997 ; MODULE:LIB$SYS_TRNLOG
05F0   998         MAC     NOVECT  LIB     LIB$SYS_TRNLOG
05F0   999
05F0  1000 ; MODULE:LIB$VM
05F0  1001         MAC     CALL    LIB     LIB$FREE_VM
05F8  1002         MAC     CALL    LIB     LIB$GET_VM
0600  1003
```

```
0600  1004 ; MODULE:LIB$STAT_VM
0600  1005         MAC     CALL    LIB     LIB$SHOW_VM
0608  1006
0608  1007 ; MODULE:LIB$CURRENCY
0608  1008         MAC     NOVECT  LIB     LIB$CURRENCY
0608  1009
0608  1010 ; MODULE:LIB$DIGIT_SEP
0608  1011         MAC     NOVECT  LIB     LIB$DIGIT_SEP
0608  1012
0608  1013 ; MODULE:LIB$RADIX_POINT
0608  1014         MAC     NOVECT  LIB     LIB$RADIX_POINT
0608  1015
0608  1016 ; MODULE:LIB$RUN_PROGRAM
0608  1017         MAC     NOVECT  LIB     LIB$RUN_PROGRAM
0608  1018
0608  1019 ; MODULE:LIB$DO_COMMAND
0608  1020         MAC     NOVECT  LIB     LIB$DO_COMMAND
0608  1021
0608  1022 ; MODULE:LIB$COMMON
0608  1023         MAC     NOVECT  LIB     LIB$GET_COMMON
0608  1024         MAC     NOVECT  LIB     LIB$PUT_COMMON
0608  1025
0608  1026 ; MODULE:LIB$TRA_ASC_EBC
0608  1027         MAC     NOVECT  LIB     LIB$TRA_ASC_EBC
0608  1028
0608  1029 ; MODULE:LIB$TRA_EBC_ASC
0608  1030         MAC     NOVECT  LIB     LIB$TRA_EBC_ASC
0608  1031
0608  1032 ; MODULE:LIB$INSQHI
0608  1033         MAC     NOVECT  LIB     LIB$INSQHI
0608  1034
0608  1035 ; MODULE:LIB$INSQTI
0608  1036         MAC     NOVECT  LIB     LIB$INSQTI
0608  1037
0608  1038 ; MODULE:LIB$REMQHI
0608  1039         MAC     NOVECT  LIB     LIB$REMQHI
0608  1040
0608  1041 ; MODULE:LIB$REMQTI
0608  1042         MAC     NOVECT  LIB     LIB$REMQTI
0608  1043
```

```
0608  1045 ;+
0608  1046 ; Internal entry points which need vectors because the non-shared
0608  1047 ; library must call these procedures in shared library, rather
0608  1048 ; than getting a copy of the procedure from the object library.
0608  1049 ; Note: the instances of $$ entry vectors is to be minimized.
0608  1050 ; The only cases where it hurts to have two copies of a procedure
0608  1051 ; is when the procedure has statically allocated (OWN) data
0608  1052 ; which is used as a process-wide resource.
0608  1053 ; Note: in order to prevent linker data truncation errors, all modules
0608  1054 ; which are shared and are also linked in as private copies when
0608  1055 ; $$entry points are referenced by non-shared routines called by the user
0608  1056 ; must declare external references to code as general (not word displacement)
0608  1057 ; even if the reference is to the same PSECT!!!!!
0608  1058 ; Modules which have this dual life are: FOR$$ERROR, FOR$$VM, FOR$$SIGNAL.
0608  1059 ;*****************************************************************
0608  1060 ; MAINTENANCE NOTE: The following $$ entry vectors can not have their
0608  1061 ; specs changed, even though $$, since that would cause user programs
0608  1062 ; with compatibility (unshared) routines to have to re-link in order to
0608  1063 ; work correctly.  Worse we would not want to increase the major ID in
0608  1064 ; order for the image activator to catch the incompatibility, since
0608  1065 ; that would cause all users to have to relink.
0608  1066 ;-
0608  1067
0608  1068 ; MODULE:FOR$$CB
0608  1069         MAC     JSB     FOR     FOR$$CB_PUSH
0610  1070         MAC     JSB     FOR     FOR$$CB_POP
0618  1071         MAC     JSB     FOR     FOR$$CB_RET
0620  1072         MAC     JSB     FOR     FOR$$CB_GET     ; Added for non-shared code
0628  1073                                                 ; to load CCB from OTS$$A_CUR_LUB
0628  1074
0628  1075
0628  1076 ; MODULE:FOR$ERRSNS                             ; See also above FOR$ERRSNS, FOR$ERRSNS_W
0628  1077         MAC     CALL    FOR     FOR$$ERRSNS_SAV
0630  1078
```

```
0630  1080 ;+
0630  1081 ;       Here starts all new entry points defined after VMS 1.00.
0630  1082 ;       Unless a FUTURE above can be replaced, all new transfer
0630  1083 ;       points MUST be appended to the end of this list!
0630  1084 ;-
0630  1085
0630  1086         MAC     CALL    FOR     FOR$IO_DC_V     ; by ref above
0638  1087         MAC     CALL    FOR     FOR$IO_GC_V
0640  1088
0640  1089 ; MODULE FOR$CVTRT
0640  1090         MAC     CALL    FOR     FOR$CVT_G_TD
0648  1091         MAC     CALL    FOR     FOR$CVT_G_TE
0650  1092         MAC     CALL    FOR     FOR$CVT_G_TF
0658  1093         MAC     CALL    FOR     FOR$CVT_G_TG
0660  1094
0660  1095 ; MODULE OTS$CVTTR
0660  1096         MAC     CALL    OTS     OTS$CVT_T_G
0668  1097
0668  1098 ; MODULE FOR$CVTRT
0668  1099         MAC     CALL    FOR     FOR$CVT_H_TD
0670  1100         MAC     CALL    FOR     FOR$CVT_H_TE
0678  1101         MAC     CALL    FOR     FOR$CVT_H_TF
0680  1102         MAC     CALL    FOR     FOR$CVT_H_TG
0688  1103
0688  1104 ; MODULE OTS$CVTTR
0688  1105         MAC     CALL    OTS     OTS$CVT_T_H
0690  1106
0690  1107 ; MODULE OTS$CVTLT       - Old entry points under FOR$
0690  1108         MAC     CALL    OTS     OTS$CVT_L_TI
0698  1109         MAC     CALL    OTS     OTS$CVT_L_TO
06A0  1110         MAC     CALL    OTS     OTS$CVT_L_TZ
06A8  1111         MAC     CALL    OTS     OTS$CVT_L_TL
06B0  1112
06B0  1113 ; MODULE FOR$ENTRY continued from above
06B0  1114         MAC     CALL    FOR     FOR$REWRITE_SF  FOR$$IO_BEG
06B8  1115         MAC     CALL    FOR     FOR$REWRITE_SO  FOR$$IO_BEG
06C0  1116         MAC     CALL    FOR     FOR$REWRITE_SU  FOR$$IO_BEG
06C8  1117         MAC     CALL    FOR     FOR$READ_IF     FOR$$IO_BEG
C6D0  1118         MAC     CALL    FOR     FOR$READ_IO     FOR$$IO_BEG
06D8  1119         MAC     CALL    FOR     FOR$WRITE_IF    FOR$$IO_BEG
06E0  1120         MAC     CALL    FOR     FOR$WRITE_IO    FOR$$IO_BEG
06E8  1121
06E8  1122 ; MODULE FOR$DELETE
06E8  1123         MAC     CALL    FOR     FOR$DELETE
06F0  1124         MAC     CALL    FOR     FOR$DELETE_D
06F8  1125
06F8  1126 ; MODULE FOR$INQUIRE
06F8  1127         MAC     CALL    FOR     FOR$INQUIRE
0700  1128
0700  1129 ; MODULE FOR$UNLOCK
0700  1130         MAC     CALL    FOR     FOR$UNLOCK
0708  1131
0708  1132 ; MODULE FOR$ENTRY continued
0708  1133         MAC     CALL    FOR     FOR$READ_KU     FOR$$IO_BEG
0710  1134
0710  1135 ; MODULE FOR$LEX
0710  1136         MAC     NOVECT  FOR     FOR$LGE
```

```
0710  1137          MAC     NOVECT  FOR     FOR$LGT
0710  1138          MAC     NOVECT  FOR     FOR$LLE
0710  1139          MAC     NOVECT  FOR     FOR$LLT
0710  1140
0710  1141 ; MODULE LIB$ADDX
0710  1142          MAC     NOVECT  LIB     LIB$ADDX
0710  1143          MAC     NOVECT  LIB     LIB$SUBX
0710  1144
0710  1145 ; MODULE LIB$ASN_WTH_MBX
0710  1146          MAC     NOVECT  LIB     LIB$ASN_WTH_MBX
0710  1147
0710  1148 ; MODULE LIB$DAY
0710  1149          MAC     NOVECT  LIB     LIB$DAY
0710  1150
0710  1151 ; MODULE LIB$EMODF
0710  1152          MAC     NOVECT  LIB     LIB$EMODF
0710  1153
0710  1154 ; MODULE LIB$EMODD
0710  1155          MAC     NOVECT  LIB     LIB$EMODD
0710  1156
0710  1157 ; MODULE LIB$EMODG
0710  1158          MAC     NOVECT  LIB     LIB$EMODG
0710  1159
0710  1160 ; MODULE LIB$EMODH
0710  1161          MAC     NOVECT  LIB     LIB$EMODH
0710  1162
0710  1163 ; MODULE LIB$EMULATE
0710  1164          MAC     NOVECT  LIB     LIB$EMULATE
0710  1165
0710  1166 ; MODULE LIB$ESTEMU
0710  1167          MAC     NOVECT  LIB     LIB$ESTEMU
0710  1168
0710  1169 ; MODULE LIB$GET_FOREIGN
0710  1170          MAC     NOVECT  LIB     LIB$GET_FOREIGN
0710  1171
0710  1172 ; MODULE LIB$POLYF
0710  1173          MAC     NOVECT  LIB     LIB$POLYF
0710  1174
0710  1175 ; MODULE LIB$POLYD
0710  1176          MAC     NOVECT  LIB     LIB$POLYD
0710  1177
0710  1178 ; MODULE LIB$POLYG
0710  1179          MAC     NOVECT  LIB     LIB$POLYG
0710  1180
0710  1181 ; MODULE LIB$POLYH
0710  1182          MAC     NOVECT  LIB     LIB$POLYH
0710  1183
0710  1184 ; MODULE LIB$SIM_TRAP
0710  1185          MAC     NOVECT  LIB     LIB$SIM_TRAP
0710  1186
0710  1187 ; MODULE LIB$TIMER
0710  1188          MAC     NOVECT  LIB     LIB$INIT_TIMER
0710  1189          MAC     NOVECT  LIB     LIB$SHOW_TIMER
0710  1190          MAC     NOVECT  LIB     LIB$STAT_TIMER
0710  1191          MAC     NOVECT  LIB     LIB$FREE_TIMER
0710  1192
0710  1193 ; MODULE MTH$AINT
```

```
0710  1194          MAC     NOVECT  MTH     MTH$AINT_R2
0710  1195
0710  1196 ; MODULE MTH$CVTDG
0710  1197          MAC     NOVECT  MTH     MTH$CVT_D_G
0710  1198          MAC     NOVECT  MTH     MTH$CVT_G_D
0710  1199
0710  1200 ; MODULE MTH$DFLOOR
0710  1201          MAC     NOVECT  MTH     MTH$DFLOOR
0710  1202          MAC     NOVECT  MTH     MTH$DFLOOR_R3
0710  1203
0710  1204 ; MODULE MTH$DIM
0710  1205          MAC     NOVECT  MTH     MTH$GDIM
0710  1206          MAC     NOVECT  MTH     MTH$HDIM
0710  1207
0710  1208 ; MODULE MTH$DINT
0710  1209          MAC     NOVECT  MTH     MTH$DINT_R4
0710  1210
0710  1211 ; MODULE MTH$DTAN
0710  1212          MAC     JSB     MTH     MTH$DTAN_R7
0718  1213
0718  1214 ; MODULE MTH$FLOOR
0718  1215          MAC     NOVECT  MTH     MTH$FLOOR
0718  1216          MAC     NOVECT  MTH     MTH$FLOOR_R1
0718  1217
0718  1218 ; MODULE MTH$GACOS
0718  1219          MAC     NOVECT  MTH     MTH$GACOS
0718  1220          MAC     NOVECT  MTH     MTH$GACOS_R7
0718  1221
0718  1222 ; MODULE MTH$GASIN
0718  1223          MAC     NOVECT  MTH     MTH$GASIN
0718  1224          MAC     NOVECT  MTH     MTH$GASIN_R7
0718  1225
0718  1226 ; MODULE MTH$GATANH
0718  1227          MAC     NOVECT  MTH     MTH$GATANH
0718  1228
0718  1229 ; MODULE MTH$GCOSH
0718  1230          MAC     NOVECT  MTH     MTH$GCOSH
0718  1231
0718  1232 ; MODULE MTH$GEXP
0718  1233          MAC     NOVECT  MTH     MTH$GEXP
0718  1234          MAC     NOVECT  MTH     MTH$GEXP_R6
0718  1235
0718  1236 ; MODULE MTH$GFLOOR
0718  1237          MAC     NOVECT  MTH     MTH$GFLOOR
0718  1238          MAC     NOVECT  MTH     MTH$GFLOOR_R3
0718  1239
0718  1240 ; MODULE MTH$GINT
0718  1241          MAC     NOVECT  MTH     MTH$GINT
0718  1242          MAC     NOVEC   MTH     MTH$GINT_R4
0718  1243
0718  1244 ; MODULE MTH$GMAX1
0718  1245          MAC     NOVECT  MTH     MTH$GMAX1
0718  1246
0718  1247 ; MODULE MTH$GMIN1
0718  1248          MAC     NOVECT  MTH     MTH$GMIN1
0718  1249
0718  1250 ; MODULE MTH$GMOD
```

```
0718  1251          MAC     NOVECT  MTH     MTH$GMOD
0718  1252
0718  1253 ; MODULE MTH$GNINT
0718  1254          MAC     NOVECT  MTH     MTH$GNINT
0718  1255
0718  1256 ; MODULE MTH$GPROD
0718  1257          MAC     NOVECT  MTH     MTH$GPROD
0718  1258
0718  1259 ; MODULE MTH$GSIGN
0718  1260          MAC     NOVECT  MTH     MTH$GSIGN
0718  1261
0718  1262 ; MODULE MTH$GSINCOS
0718  1263          MAC     NOVECT  MTH     MTH$GSIN
0718  1264          MAC     NOVECT  MTH     MTH$GCOS
0718  1265          MAC     NOVECT  MTH     MTH$GSIN_R7
0718  1266          MAC     NOVECT  MTH     MTH$GCOS_R7
0718  1267
0718  1268 ; MODULE MTH$GSINH
0718  1269          MAC     NOVECT  MTH     MTH$GSINH
0718  1270
0718  1271 ; MODULE MTH$GSQRT
0718  1272          MAC     NOVECT  MTH     MTH$GSQRT
0718  1273          MAC     NOVECT  MTH     MTH$GSQRT_R5
0718  1274
0718  1275 ; MODULE MTH$GTAN
0718  1276          MAC     NOVECT  MTH     MTH$GTAN
0718  1277          MAC     NOVECT  MTH     MTH$GTAN_R7
0718  1278
0718  1279 ; MODULE MTH$GTANH
0718  1280          MAC     NOVECT  MTH     MTH$GTANH
0718  1281
0718  1282 ; MODULE MTH$HACOS
0718  1283          MAC     NOVECT  MTH     MTH$HACOS
0718  1284          MAC     NOVECT  MTH     MTH$HACOS_R8
0718  1285
0718  1286 ; MODULE MTH$HASIN
0718  1287          MAC     NOVECT  MTH     MTH$HASIN
0718  1288          MAC     NOVECT  MTH     MTH$HASIN_R8
0718  1289
0718  1290 ; MODULE MTH$HCOSH
0718  1291          MAC     NOVECT  MTH     MTH$HCOSH
0718  1292
0718  1293 ; MODULE MTH$HEXP
0718  1294          MAC     NOVECT  MTH     MTH$HEXP
0718  1295          MAC     NOVECT  MTH     MTH$HEXP_R6
0718  1296
0718  1297 ; MODULE MTH$HFLOOR
0718  1298          MAC     NOVECT  MTH     MTH$HFLOOR
0718  1299          MAC     NOVECT  MTH     MTH$HFLOOR_R7
0718  1300
0718  1301 ; MODULE MTH$HINT
0718  1302          MAC     NOVECT  MTH     MTH$HINT
0718  1303          MAC     NOVECT  MTH     MTH$HINT_R8
0718  1304 ; MODULE MTH$HMAX1
0718  1305          MAC     NOVECT  MTH     MTH$HMAX1
0718  1306
0718  1307 ; MODULE MTH$HMIN1
```

```
0718  1308          MAC     NOVECT  MTH     MTH$HMIN1
0718  1309
0718  1310 ; MODULE MTH$HMOD
0718  1311          MAC     NOVECT  MTH     MTH$HMOD
0718  1312
0718  1313 ; MODULE MTH$HNINT
0718  1314          MAC     NOVECT  MTH     MTH$HNINT
0718  1315
0718  1316 ; MODULE MTH$HSIGN
0718  1317          MAC     NOVECT  MTH     MTH$HSIGN
0718  1318
0718  1319 ; MODULE MTH$HSINCOS
0718  1320          MAC     NOVECT  MTH     MTH$HSIN
0718  1321          MAC     NOVECT  MTH     MTH$HSIN_R5
0718  1322          MAC     NOVECT  MTH     MTH$HCOS
0718  1323          MAC     NOVECT  MTH     MTH$HCOS_R5
0718  1324
0718  1325 ; MODULE MTH$HSINH
0718  1326          MAC     NOVECT  MTH     MTH$HSINH
0718  1327
0718  1328 ; MODULE MTH$HSQRT
0718  1329          MAC     NOVECT  MTH     MTH$HSQRT
0718  1330          MAC     NOVECT  MTH     MTH$HSQRT_R8
0718  1331
0718  1332 ; MODULE MTH$HTAN
0718  1333          MAC     NOVECT  MTH     MTH$HTAN
0718  1334          MAC     NOVECT  MTH     MTH$HTAN_R5
0718  1335
0718  1336 ; MODULE MTH$HTANH
0718  1337          MAC     NOVECT  MTH     MTH$HTANH
0718  1338
0718  1339 ; MODULE MTH$IIGNNT
0718  1340          MAC     NOVECT  MTH     MTH$IIGNNT
0718  1341
0718  1342 ; MODULE MTH$IIHNNT
0718  1343          MAC     NOVECT  MTH     MTH$IIHNNT
0718  1344
0718  1345 ; MODULE MTH$JIGNNT
0718  1346          MAC     NOVECT  MTH     MTH$JIGNNT
0718  1347
0718  1348 ; MODULE MTH$JIHNNT
0718  1349          MAC     NOVECT  MTH     MTH$JIHNNT
0718  1350
0718  1351 ; MODULE MTH$TAN
0718  1352          MAC     JSB     MTH     MTH$TAN_R4
0720  1353
0720  1354 ; MODULE MTH$SGN
0720  1355          MAC     NOVECT  MTH     MTH$SGN
0720  1356          MAC     NOVECT  MTH     MTH$SGN_R1
0720  1357
0720  1358 ; MODULE OTS$POWGG
0720  1359          MAC     NOVECT  OTS     OTS$POWGG
0720  1360
0720  1361 ; MODULE OTS$POWGJ
0720  1362          MAC     NOVECT  OTS     OTS$POWGJ
0720  1363
0720  1364 ; MODULE OTS$POWHJ
```

```
0720  1365          MAC     NOVECT  OTS     OTS$POWHJ_R3
0720  1366
0720  1367 ; MODULE OTS$DIVCD
0720  1368          MAC     NOVECT  OTS     OTS$DIVCD_R3
0720  1369
0720  1370 ; MODULE OTS$DIVCG
0720  1371          MAC     NOVECT  OTS     OTS$DIVCG_R3
0720  1372
0720  1373 ; MODULE OTS$MULCD
0720  1374          MAC     NOVECT  OTS     OTS$MULCD_R3
0720  1375
0720  1376 ; MODULE OTS$MULCG
0720  1377          MAC     NOVECT  OTS     OTS$MULCG_R3
0720  1378
0720  1379 ; MODULE MTH$CDABS
0720  1380          MAC     NOVECT  MTH     MTH$CDABS
0720  1381
0720  1382 ; MODULE MTH$DCONJG
0720  1383          MAC     NOVECT  MTH     MTH$DCONJG
0720  1384
0720  1385 ; MODULE MTH$CDEXP
0720  1386          MAC     NOVECT  MTH     MTH$CDEXP
0720  1387
0720  1388 ; MODULE MTH$CDLOG
0720  1389          MAC     NOVECT  MTH     MTH$CDLOG
0720  1390
0720  1391 ; MODULE MTH$CDSINCOS
0720  1392          MAC     NOVECT  MTH     MTH$CDCOS
0720  1393          MAC     NOVECT  MTH     MTH$CDSIN
0720  1394
0720  1395 ; MODULE MTH$CDSQRT
0720  1396          MAC     NOVECT  MTH     MTH$CDSQRT
0720  1397
0720  1398 ; MODULE MTH$CGABS
0720  1399          MAC     NOVECT  MTH     MTH$CGABS
0720  1400
0720  1401 ; MODULE MTH$GCONJG
0720  1402          MAC     NOVECT  MTH     MTH$GCONJG
0720  1403
0720  1404 ; MODULE MTH$CGEXP
0720  1405          MAC     NOVECT  MTH     MTH$CGEXP
0720  1406
0720  1407 ; MODULE MTH$CGLOG
0720  1408          MAC     NOVECT  MTH     MTH$CGLOG
0720  1409
0720  1410 ; MODULE MTH$CGSINCOS
0720  1411          MAC     NOVECT  MTH     MTH$CGCOS
0720  1412          MAC     NOVECT  MTH     MTH$CGSIN
0720  1413
0720  1414 ; MODULE MTH$CGSQRT
0720  1415          MAC     NOVECT  MTH     MTH$CGSQRT
0720  1416
0720  1417 ; MODULE OTS$POWCC
0720  1418          MAC     NOVECT  OTS     OTS$POWCC
0720  1419
0720  1420 ; MODULE OTS$POWCDCD
0720  1421          MAC     NOVECT  OTS     OTS$POWCDCD_R3
```

```
0720  1422
0720  1423 ; MODULE OTS$POWCDJ
0720  1424         MAC     NOVECT  OTS     OTS$POWCDJ_R3
0720  1425
0720  1426 ; MODULE OTS$POWCGCG
0720  1427         MAC     NOVECT  OTS     OTS$POWCGCG_R3
0720  1428
0720  1429 ; MODULE OTS$POWCGJ
0720  1430         MAC     NOVECT  OTS     OTS$POWCGJ_R3
0720  1431
0720  1432 ; MODULE OTS$POWHH
0720  1433         MAC     NOVECT  OTS     OTS$POWHH_R3
0720  1434
0720  1435 ; MODULE MTH$SQRT
0720  1436         MAC     JSB     MTH     MTH$SQRT_R3
0728  1437
0728  1438 ;+
0728  1439 ; The following routine is called from FOR$$ERROR, therefore
0728  1440 ; it must be vectored.
0728  1441 ;-
0728  1442
0728  1443 ; MODULE FOR$$CB (continued)
0728  1444         MAC     CALL    FOR     FOR$$FP_MATCH
0730  1445
0730  1446 ; MODULE FOR$READ_SN
0730  1447         MAC     CALL    FOR     FOR$READ_SN     FOR$$IO_BEG
0738  1448
0738  1449 ; MODULE FOR$WRITE_SN
0738  1450         MAC     CALL    FOR     FOR$WRITE_SN    FOR$$IO_BEG
0740  1451
0740  1452 ; MODULE FOR$IO_ELEM (continued)
0740  1453         MAC     CALL    FOR     FOR$IO_X_SB
0748  1454         MAC     CALL    FOR     FOR$IO_X_NL
0750  1455         MAC     CALL    FOR     FOR$IO_X_SE
0758  1456
0758  1457 ; MODULE OTS$CVTLT (continued)
0758  1458         MAC     CALL    OTS     OTS$CVT_L_TB
0760  1459
0760  1460 ; MODULE OTS$CVTTOL (continued)
0760  1461         MAC     CALL    OTS     OTS$CVT_TB_L
0768  1462
0768  1463 ; MODULE OTS$CVTTF
0768  1464         MAC     CALL    OTS     OTS$CVT_T_F
0770  1465
0770  1466 ; MODULE LIB$ATTACH
0770  1467         MAC     CALL    LIB     LIB$ATTACH
0778  1468
0778  1469 ; MODULE LIB$SPAWN
0778  1470         MAC     CALL    LIB     LIB$SPAWN
0780  1471
0780  1472 ; MODULE LIB$GET_OPCODE
0780  1473         MAC     CALL    LIB     LIB$GET_OPCODE
0788  1474
0788  1475 ; MODULE FOR$RAB
0788  1476         MAC     CALL    FOR     FOR$RAB
0790  1477
0790  1478 ;+
```

```
                     0790  1479 ; The following three entry points are for the "kernel" floating output
                     0790  1480 ; conversion routines.  Although they are "double-dollar" names, they are
                     0790  1481 ; vectored so that future language-specific shareable images can use them.
                     0790  1482 ;-
                     0790  1483
                     0790  1484 ; MODULE OTS$$CVTDT
                     0790  1485         MAC     JSB     OTS     OTS$$CVT_D_T_R8
                     0798  1486
                     0798  1487 ; MODULE OTS$$CVTRT
                     0798  1488         MAC     JSB     OTS     OTS$$CVT_G_T_R8
                     07A0  1489         MAC     JSB     OTS     OTS$$CVT_H_T_R8
                     07A8  1490
                     07A8  1491 ;+
                     07A8  1492 ; The following entries are present only so that there will be references
                     07A8  1493 ; to these symbols in this module.  Without them, the linker complains.
                     07A8  1494 ; Note that these entries are not universal, so they cannot be referenced
                     07A8  1495 ; by those linking to this image.
                     07A8  1496 ;-
                     07A8  1497
               0000' 07A8  1498         .MASK   BAS$$HANDLER
00000000'GF      17  07AA  1499         JMP     G^BAS$$HANDLER
               0000' 07B0  1500         .MASK   COB$$HANDLER
00000000'GF      17  07B2  1501         JMP     G^COB$$HANDLER
               0000' 07B8  1502         .MASK   FOR$$IO_BEG
00000000'GF      17  07BA  1503         JMP     G^FOR$$IO_BEG
                     07C0  1504
```

```
                                    07C0  1506 ;+
                                    07C0  1507 ; Leave some expansion room for FOR, OTS and MTH by padding to
                                    07C0  1508 ; a page boundry.  If this room is exhausted they will have to go after
                                    07C0  1509 ; the STR and BAS facilities.  Give an error message if the space gets
                                    07C0  1510 ; exhausted.
                                    07C0  1511 ;-
                           FFFFFFC0 07C0  1512 .IF GREATER <<.-RTL$START>-^X800>
                                    07C0  1513 .ERROR <<.-RTL$START>-^X800>    ; Negative vector pad space
                                    07C0  1514 .ENDC
00'00'00'00'00'00'00'00'00'00'00'00' 07C0  1515         .BYTE   0[512-<<.-RTL$START>&511>]
00'00'00'00'00'00'00'00'00'00'00'00' 07CC
00'00'00'00'00'00'00'00'00'00'00'00' 07D8
00'00'00'00'00'00'00'00'00'00'00'00' 07E4
00'00'00'00'00'00'00'00'00'00'00'00' 07F0
                     00'00'00'00'    07FC
```

```
0800  1517 ;+
0800  1518 ; The following entry points are for the string library.
0800  1519 ; The JSB entry points are later.
0800  1520 ;-
0800  1521          MAC     CALL    STR     STR$CONCAT              ; Concatenate strings
0808  1522          MAC     CALL    STR     STR$COPY_DX             ; Copy by descriptor
0810  1523          MAC     CALL    STR     STR$COPY_R              ; Copy by reference
0818  1524          MAC     CALL    STR     STR$FREE1_DX            ; Free a string
0820  1525          MAC     CALL    STR     STR$GET1_DX             ; Allocate a string
0828  1526          MAC     NOVECT  STR     STR$ADD                 ; Add two strings
0828  1527          MAC     NOVECT  STR     STR$MUL                 ; Multiply two strings
0828  1528          MAC     NOVECT  STR     STR$RECIP               ; Take 1/ a string
0828  1529          MAC     NOVECT  STR     STR$ROUND               ; Arithmeticly round a strin
0828  1530          MAC     CALL    STR     STR$LEFT                ; Take left part of string
0830  1531          MAC     NOVECT  STR     STR$LEFT_R8             ; (JSB entry point)
0830  1532          MAC     CALL    STR     STR$LEN_EXTR            ; Extract from a string by l
0838  1533          MAC     NOVECT  STR     STR$LEN_EXTR_R8         ; (JSB entry point)
0838  1534          MAC     CALL    STR     STR$POS_EXTR            ; Extract from a string by p
0840  1535          MAC     NOVECT  STR     STR$POS_EXTR_R8         ; (JSB entry point)
0840  1536          MAC     CALL    STR     STR$POSITION            ;
0848  1537          MAC     NOVECT  STR     STR$POSITION_R6         ; (JSB entry point)
0848  1538          MAC     CALL    STR     STR$RIGHT               ; Take right part of a strin
0850  1539          MAC     NOVECT  STR     STR$RIGHT_R8            ; (JSB entry point)
0850  1540          MAC     CALL    STR     STR$DUPL_CHAR           ; Make lots of a character
0858  1541          MAC     CALL    STR     STR$TRIM                ; Remove trailing blanks
0860  1542          MAC     FUTURE  STR     STR$FUTURE_1            ; Reserved for future expans
0868  1543          MAC     FUTURE  STR     STR$FUTURE_2
0870  1544          MAC     FUTURE  STR     STR$FUTURE_3
0878  1545 ;+
0878  1546 ; String condition codes
0878  1547 ;-
0878  1548          MAC     SYM     STR     STR$_DIVBY_ZER          ; Divide by zero
0878  1549          MAC     SYM     STR     STR$_FATINTERR          ; Fatal internal error
0878  1550          MAC     SYM     STR     STR$_ILLSTRCLA          ; Illegal string class
0878  1551          MAC     SYM     STR     STR$_ILLSTRPOS
0878  1552          MAC     SYM     STR     STR$_ILLSTRSPE
0878  1553          MAC     SYM     STR     STR$_INSVIRMEM
0878  1554          MAC     SYM     STR     STR$_MATCH
0878  1555          MAC     SYM     STR     STR$_NEGSTRLEN
0878  1556          MAC     SYM     STR     STR$_NOMATCH
0878  1557          MAC     SYM     STR     STR$_NORMAL
0878  1558          MAC     SYM     STR     STR$_STRIS_INT
0878  1559          MAC     SYM     STR     STR$_STRTOOLON
0878  1560          MAC     SYM     STR     STR$_TRU
0878  1561          MAC     SYM     STR     STR$_WRONUMARG
0878  1562
```

```
0878  1564 ;+
0878  1565 ; The following entry points are generated by the BASIC-PLUS-2
0878  1566 ; compiler.  The current arrangement of which are vectored is tenative.
0878  1567 ;-
0878  1568 ;
0878  1569 ;       ARITHMETIC CODE SUPPORT
0878  1570 ;
0878  1571         MAC     NOVECT  BAS     BAS$POWII               ;  Integer(w) ** Integer(w)
0878  1572         MAC     JSB     BAS     BAS$SCALE_D_R1          ;  Scale a number
0880  1573         MAC     NOVECT  BAS     BAS$POWJJ               ;  Long ** Long
0880  1574         MAC     JSB     BAS     BAS$DSCALE_D_R1         ;  Descale a number
0888  1575         MAC     NOVECT  BAS     BAS$POWRJ               ;  Float ** Long
0888  1576         MAC     NOVECT  BAS     BAS$POWRR               ;  Float ** Float
0888  1577         MAC     NOVECT  BAS     BAS$POWDJ               ;  Double ** Long
0888  1578         MAC     NOVECT  BAS     BAS$POWDD               ;  Double ** Double
0888  1579         MAC     NOVECT  BAS     BAS$RND_F_R1            ;  Return random number
0888  1580         MAC     NOVECT  BAS     BAS$RANDOMIZE           ;  Perturb the random seed
0888  1581         MAC     CALL    BAS     BAS$CMPF_APP            ;  Approximate float compare
0890  1582         MAC     CALL    BAS     BAS$CMPD_APP            ;  Approximate double compar
0898  1583 ;
0898  1584 ;       STRING CODE SUPPORT
0898  1585 ;
0898  1586         MAC     NOVECT  BAS     BAS$CHANGE_NA_S         ;  CHANGE A% to A$
0898  1587         MAC     NOVECT  BAS     BAS$CHANGE_S_NA         ;  CHANGE A$ to A%
```

```
0898  1589 ;
0898  1590 ;       STRING FUNCTIONS
0898  1591 ;
0898  1592          MAC     CALL    BAS     BAS$RSET                ;  String move, right justif
08A0  1593          MAC     CALL    BAS     BAS$RSET_R              ;  By-ref entry point
08A8  1594          MAC     CALL    BAS     BAS$EDIT                ;  String editing
08B0  1595          MAC     NOVECT  BAS     BAS$FORMAT_F            ;  Floating FORMAT$
08B0  1596          MAC     NOVECT  BAS     BAS$FORMAT_D            ;  Double FORMAT$
08B0  1597          MAC     CALL    BAS     BAS$INSTR               ;  Match substring
08B8  1598          MAC     NOVECT  BAS     BAS$RAD                 ;  RADIX 50
08B8  1599          MAC     NOVECT  BAS     BAS$RAD50               ;  RADIX 50
08B8  1600          MAC     NOVECT  BAS     BAS$STOP                ;  STOP statement (*** here
08B8  1601          MAC     CALL    BAS     BAS$STR_F               ;  Return binary->ASCII numb
08C0  1602          MAC     CALL    BAS     BAS$STR_D               ;  Ditto for double
08C8  1603          MAC     CALL    BAS     BAS$STR_L               ;  Ditto for longword
08D0  1604          MAC     CALL    BAS     BAS$NUM_F               ;  Return binary->ASCII numb
08D8  1605          MAC     CALL    BAS     BAS$NUM_D               ;  Ditto for double
08E0  1606          MAC     CALL    BAS     BAS$NUM_L               ;  Ditto for longword
08E8  1607          MAC     CALL    BAS     BAS$NUMT_F              ;  Return binary->ASCII numb
08F0  1608          MAC     CALL    BAS     BAS$NUM1_D              ;  Ditto for double
08F8  1609          MAC     CALL    BAS     BAS$NUM1_L              ;  Ditto for longword
0900  1610          MAC     NOVECT  BAS     BAS$TAB                 ;  Tab over x spaces
0900  1611          MAC     NOVECT  BAS     BAS$TIME_T              ;  24 hour time string
0900  1612          MAC     CALL    BAS     BAS$VAL_L               ;  Return ASCII->binary stri
0908  1613          MAC     CALL    BAS     BAS$VAL_F               ;  Ditto for floating
0910  1614          MAC     CALL    BAS     BAS$VAL_D               ;  Ditto for double
0918  1615          MAC     NOVECT  BAS     BAS$XLATE               ;  Translate a string
```

```
0918  1617 ;
0918  1618 ;       STRING ARITHMETIC
0918  1619 ;
0918  1620          MAC     NOVECT  BAS     BAS$COMP                ; String arith compare
0918  1621          MAC     NOVECT  BAS     BAS$DIF                 ; S.A. difference
0918  1622          MAC     NOVECT  BAS     BAS$PLACE               ; S.A. precision
0918  1623          MAC     NOVECT  BAS     BAS$PROD                ; S.A. multiplication
0918  1624          MAC     NOVECT  BAS     BAS$QUO                 ; S.A. division
0918  1625          MAC     NOVECT  BAS     BAS$SUM                 ; S.A. addition
0918  1626 ;
0918  1627 ;       PROCEDURE ACTIVATION
0918  1628 ;
0918  1629          MAC     JSB     BAS     BAS$INIT_R8             ; Main program initializer
0920  1630          MAC     JSB     BAS     BAS$INIT_DEF_R8         ; DEF function initializer
0928  1631          MAC     JSB     BAS     BAS$INIT_DFS_R8         ; DEF* function initializer
0930  1632          MAC     CALL    BAS     BAS$INIT_GOSUB          ; GOSUB initializer
0938  1633          MAC     NOVECT  BAS     BAS$INIT_C_GSB          ; ON-GOSUB initializer
0938  1634          MAC     JSB     BAS     BAS$END_R8              ; Main program ender
0940  1635          MAC     JSB     BAS     BAS$END_DEF_R8          ; DEF function end
0948  1636          MAC     JSB     BAS     BAS$END_DFS_R8          ; DEF* function end
0950  1637          MAC     JSB     BAS     BAS$END_GSB_R8          ; GOSUB end
0958  1638 ;
0958  1639 ;       ERROR HANDLING
0958  1640 ;
0958  1641          MAC     CALL    BAS     BAS$ON_ERR_Z            ; ON ERROR GOTO 0
0960  1642          MAC     CALL    BAS     BAS$ON_ERR_BK           ; ON ERROR GOBACK
0968  1643          MAC     CALL    BAS     BAS$RESUME              ; RESUME line num
0970  1644          MAC     CALL    BAS     BAS$RESUME_Z            ; RESUME
0978  1645          MAC     CALL    BAS     BAS$ERR                 ; ERR variable
0980  1646          MAC     CALL    BAS     BAS$ERL                 ; ERL variable
0988  1647          MAC     CALL    BAS     BAS$ERN                 ; ERN$ variable
0990  1648          MAC     CALL    BAS     BAS$ERT                 ; ERT$ variable
0998  1649
0998  1650          MAC     CALL    BAS     BAS$HANDLER     BAS$$HANDLER
```

```
09A0  1652 ;
09A0  1653 ;       Scalar (non matrix) I/O
09A0  1654 ;
09A0  1655         MAC     CALL    BAS     BAS$INPUT               ;  Initialize for INPUT unit
09A8  1656         MAC     CALL    BAS     BAS$LINPUT              ;  Ditto LINPUT unit
09B0  1657         MAC     CALL    BAS     BAS$INPUT_LINE          ;  Ditto INPUT LINE unit
09B8  1658         MAC     CALL    BAS     BAS$READ                ;  Ditto READ
09C0  1659         MAC     CALL    BAS     BAS$PRINT               ;  Ditto PRINT
09C8  1660         MAC     CALL    BAS     BAS$PRINT_USING         ;  Ditto PRINT USING
09D0  1661         MAC     CALL    BAS     BAS$IO_END              ;  End of I/O list
09D8  1662         MAC     CALL    BAS     BAS$IN_W_R              ;  INPUT word
09E0  1663         MAC     CALL    BAS     BAS$IN_L_R              ;  INPUT long
09E8  1664         MAC     CALL    BAS     BAS$IN_F_R              ;  INPUT float
09F0  1665         MAC     CALL    BAS     BAS$IN_D_R              ;  INPUT double
09F8  1666         MAC     CALL    BAS     BAS$IN_T_DX             ;  INPUT string
0A00  1667         MAC     CALL    BAS     BAS$OUT_L_V_S           ;  PRINT long(word) ;
0A08  1668         MAC     CALL    BAS     BAS$OUT_L_V_B           ;  PRINT long(word)
0A10  1669         MAC     CALL    BAS     BAS$OUT_L_V_C           ;  PRINT long(word) ,
0A18  1670         MAC     CALL    BAS     BAS$OUT_F_V_S           ;  PRINT float ;
0A20  1671         MAC     CALL    BAS     BAS$OUT_F_V_B           ;  PRINT float
0A28  1672         MAC     CALL    BAS     BAS$OUT_F_V_C           ;  PRINT float ,
0A30  1673         MAC     CALL    BAS     BAS$OUT_D_V_S           ;  PRINT double ;
0A38  1674         MAC     CALL    BAS     BAS$OUT_D_V_B           ;  PRINT double
0A40  1675         MAC     CALL    BAS     BAS$OUT_D_V_C           ;  PRINT double ,
0A48  1676         MAC     CALL    BAS     BAS$OUT_T_DX_S          ;  PRINT string ;
0A50  1677         MAC     CALL    BAS     BAS$OUT_T_DX_B          ;  PRINT string
0A58  1678         MAC     CALL    BAS     BAS$OUT_T_DX_C          ;  PRINT string ,
0A60  1679 ;
0A60  1680 ;  Matrix I/O
0A60  1681 ;
0A60  1682         MAC     NOVECT  BAS     BAS$OUT_MAT_S           ;  Output element xmtr
0A60  1683         MAC     NOVECT  BAS     BAS$OUT_MAT_B           ;
0A60  1684         MAC     NOVECT  BAS     BAS$OUT_MAT_C           ;
0A60  1685         MAC     NOVECT  BAS     BAS$IN_MAT              ;  Input element xmtr
0A60  1686         MAC     CALL    BAS     BAS$MAT_PRINT           ;  Init for MAT PRINT
0A68  1687         MAC     CALL    BAS     BAS$MAT_INPUT           ;       for MAT INPUT
0A70  1688         MAC     CALL    BAS     BAS$MAT_LINPUT          ;       for MAT LINPUT
0A78  1689         MAC     CALL    BAS     BAS$MAT_READ            ;       for MAT READ
0A80  1690         MAC     NOVECT  BAS     BAS$NUM                 ;  NUM var (mat)
0A80  1691         MAC     NOVECT  BAS     BAS$NUM2                ;  NUM2 var (mat)
0A80  1692
0A80  1693 ;
0A80  1694 ;       RMS I/O
0A80  1695 ;
0A80  1696         MAC     CALL    BAS     BAS$OPEN                ;  OPEN (all)
0A88  1697         MAC     CALL    BAS     BAS$CLOSE               ;  CLOSE (all)
0A90  1698         MAC     CALL    BAS     BAS$GET                 ;  Sequential GET
0A98  1699         MAC     CALL    BAS     BAS$GET_RECORD          ;  Random GET
0AA0  1700         MAC     CALL    BAS     BAS$GET_KEY             ;  Indexed GET
0AA8  1701         MAC     CALL    BAS     BAS$PUT                 ;  Sequential PUT
0AB0  1702         MAC     CALL    BAS     BAS$PUT_RECORD          ;  Random PUT
0AB8  1703         MAC     CALL    BAS     BAS$PUT_COUNT           ;  Sequential PUT w/COUNT
0AC0  1704         MAC     CALL    BAS     BAS$PUT_REC_CNT         ;  Random PUT w/COUNT
0AC8  1705         MAC     CALL    BAS     BAS$FIND                ;  Sequential FIND
0AD0  1706         MAC     CALL    BAS     BAS$FIND_RECORD         ;  Random FIND
0AD8  1707         MAC     CALL    BAS     BAS$FIND_KEY            ;  Indexed FIND
0AE0  1708         MAC     CALL    BAS     BAS$DELETE              ;  DELETE
```

```
0AE8  1709          MAC     CALL    BAS     BAS$UPDATE              ;  UPDATE
0AF0  1710          MAC     CALL    BAS     BAS$UPDATE_COUN         ;  UPDATE w/COUNT
0AF8  1711          MAC     CALL    BAS     BAS$RESTORE             ;  RESTORE
0B00  1712          MAC     CALL    BAS     BAS$RESTORE_KEY         ;  Indexed RESTORE
0B08  1713          MAC     CALL    BAS     BAS$SCRATCH             ;  SCRATCH
0B10  1714          MAC     CALL    BAS     BAS$UNLOCK              ;  Release
0B18  1715          MAC     CALL    BAS     BAS$FREE                ;  FREE
```

```
                     ;
                     ;       RSTS/E COMPATIBILTY
                     ;
0B20  1720           MAC     NOVECT  BAS     BAS$CVT_W_S             ; CVT%$
0B20  1721           MAC     NOVECT  BAS     BAS$CVT_S_W             ; CVT$%
0B20  1722           MAC     NOVECT  BAS     BAS$CVT_F_S             ; CVTF$
0B20  1723           MAC     NOVECT  BAS     BAS$CVT_D_S             ; CVTD$
0B20  1724           MAC     NOVECT  BAS     BAS$CVT_S_F             ; CVT$F
0B20  1725           MAC     NOVECT  BAS     BAS$CVT_S_D             ; CVT$D
0B20  1726           MAC     NOVECT  BAS     BAS$FSS                 ; File string scan
0B20  1727           MAC     NOVECT  BAS     BAS$FSP                 ; File info
0B20  1728           MAC     NOVECT  BAS     BAS$SYS                 ; Sys calls
0B20  1729           MAC     NOVECT  BAS     BAS$CHAIN               ; CHAIN statement
0B20  1730           MAC     NOVECT  BAS     BAS$PEEK                ; Examine RSTS/E memory
0B20  1731           ;
0B20  1732           ;       MISC
0B20  1733           ;
0B20  1734           MAC     CALL    BAS     BAS$CCPOS               ; CCPOS func
0B28  1735           MAC     NOVECT  BAS     BAS$ECHO                ; Enable echo
0B28  1736           MAC     NOVECT  BAS     BAS$ONECHR              ; *********
0B28  1737           MAC     NOVECT  BAS     BAS$NOECHO              ; Disable echo
0B28  1738           MAC     NOVECT  BAS     BAS$RCTRLO              ; Disable CTRL O
0B28  1739           MAC     NOVECT  BAS     BAS$CTRLO               ; Cause control O
0B28  1740           MAC     CALL    BAS     BAS$RESTORE_DAT         ; RESTORE data
0B30  1741           MAC     CALL    BAS     BAS$RECOUNT             ; RECOUNT var
0B38  1742           MAC     CALL    BAS     BAS$STATUS              ; STATUS var
0B40  1743           MAC     NOVECT  BAS     BAS$MAGTAPE             ; MAGTAPE func
0B40  1744           MAC     NOVECT  BAS     BAS$TIME_F              ; Floating time values
0B40  1745           MAC     NOVECT  BAS     BAS$SLEEP               ; SLEEP statement
0B40  1746           MAC     NOVECT  BAS     BAS$NAME_AS             ; NAME AS statement
0B40  1747           MAC     NOVECT  BAS     BAS$KILL                ; KILL erase
0B40  1748           MAC     CALL    BAS     BAS$BUFSIZ              ; BUFSIZ func
0B48  1749           ;
0B48  1750           ;       MOVE STATEMENT
0B48  1751           ;
0B48  1752           MAC     NOVECT  BAS     BAS$MOVE_TO             ; Start a MOVE TO statement
0B48  1753           MAC     NOVECT  BAS     BAS$MOVE_FROM           ; Start a MOVE FROM statemen
0B48  1754           MAC     NOVECT  BAS     BAS$MOVE_END            ; End of a MOVE statement
0B48  1755           MAC     NOVECT  BAS     BAS$MOVE_ARRAY          ; MOVE an array
0B48  1756
```

```
0B48  1758 ;
0B48  1759 ;       MATRIX Arithmetic
0B48  1760 ;
0B48  1761         MAC     NOVECT  BAS     BAS$DET_F                       ;  Determinate
0B48  1762         MAC     NOVECT  BAS     BAS$DET_D                       ;  Determinate
0B48  1763         MAC     NOVECT  BAS     BAS$MAT_NULL                    ;  Null out string matrix
0B48  1764         MAC     NOVECT  BAS     BAS$MAT_ASSIGN                  ;  Matrix assignments
0B48  1765         MAC     NOVECT  BAS     BAS$MAT_INIT                    ;  Matrix initialize(0 or 1)
0B48  1766         MAC     NOVECT  BAS     BAS$MAT_IDN                     ;  Matrix identity
0B48  1767         MAC     NOVECT  BAS     BAS$MAT_ADD                     ;  Matrix addition
0B48  1768         MAC     NOVECT  BAS     BAS$MAT_SUB                     ;  Matrix subtraction
0B48  1769         MAC     NOVECT  BAS     BAS$MAT_MUL                     ;  Matrix multiplication
0B48  1770         MAC     NOVECT  BAS     BAS$MAT_SCA_MUL                 ;  Matrix scalar multiplicat
0B48  1771         MAC     NOVECT  BAS     BAS$MAT_TRN                     ;  Matrix transposition
0B48  1772         MAC     NOVECT  BAS     BAS$MAT_INV                     ;  Matrix inversion
0B48  1773         MAC     NOVECT  BAS     BAS$MAT_REDIM                   ;  Single redimension
0B48  1774 ;
0B48  1775 ;       CODE SUPPORT
0B48  1776 ;
0B48  1777         MAC     CALL    BAS     BAS$CHR                         ;  Return character for bina
0B50  1778 ;
0B50  1779 ;       VIRTUAL ARRAYS
0B50  1780 ;
0B50  1781         MAC     NOVECT  BAS     BAS$FET_FA_W_R8                 ;  Fetch a word from virtual
0B50  1782         MAC     NOVECT  BAS     BAS$FET_FA_L_R8
0B50  1783         MAC     NOVECT  BAS     BAS$FET_FA_F_R8
0B50  1784         MAC     NOVECT  BAS     BAS$FET_FA_D_R8
0B50  1785         MAC     NOVECT  BAS     BAS$FETCH_BFA
0B50  1786         MAC     NOVECT  BAS     BAS$STO_FA_W_R8                 ;  Store a word in a virtual
0B50  1787         MAC     NOVECT  BAS     BAS$STO_FA_L_R8
0B50  1788         MAC     NOVECT  BAS     BAS$STO_FA_F_R8
0B50  1789         MAC     NOVECT  BAS     BAS$STO_FA_D_R8
0B50  1790         MAC     NOVECT  BAS     BAS$STORE_BFA
0B50  1791         MAC     NOVECT  BAS     BAS$STO_FA_RDX
```

```
0B50  1793 ;
0B50  1794 ;       FIELD STATEMENT
0B50  1795 ;
0B50  1796         MAC     NOVECT  BAS     BAS$FIELD_SET           ; Set up FIELD var
0B50  1797         MAC     NOVECT  BAS     BAS$FIELD_COPY          ; Copy a FIELDed var
0B50  1798         MAC     NOVECT  BAS     BAS$FIELD_CLEAR         ;  Clear the fielded attribu
0B50  1799         MAC     NOVECT  BAS     BAS$FIELD_PURGE         ; ?
0B50  1800         MAC     NOVECT  BAS     BAS$FIELD_OPEN          ; ?
0B50  1801         MAC     NOVECT  BAS     BAS$FIELD_CLOSE         ; ?
0B50  1802 ;
0B50  1803 ;       MISC
0B50  1804 ;
0B50  1805         MAC     NOVECT  BAS     BAS$DATE_T              ;  Return an ASCII string w/
0B50  1806         MAC     CALL    BAS     BAS$ERROR               ;  Signal errors from compil
0B58  1807         MAC     NOVECT  LIB     LIB$DATE_TIME           ;  System standard date/time
0B58  1808         MAC     NOVECT  BAS     BAS$MARGIN              ;  MARGIN sta/MAR% funct
0B58  1809         MAC     NOVECT  BAS     BAS$NOMARGIN            ;  NOMARGIN statement
0B58  1810 ;
0B58  1811 ;       LIB$TPARSE and its subroutines
0B58  1812 ;
0B58  1813         MAC     CALL    LIB     LIB$TPARSE
0B60  1814         MAC     NOVECT  LIB     LIB$CVT_DTB
0B60  1815         MAC     NOVECT  LIB     LIB$CVT_HTB
0B60  1816         MAC     NOVECT  LIB     LIB$CVT_OTB
```

```
                        ;
                        ; Entry points used by the BASIC compiler to support the RUN command.
                        ;
                        MAC     NOVECT  BAS     BAS$RUN_INIT            ; Initialize for RUN
                        MAC     CALL    BAS     BAS$PUSH_ERR            ; Save error status
                        MAC     CALL    BAS     BAS$POP_ERR             ; Restore error status
                        MAC     NOVECT  BAS     BAS$INIT_IOL            ; Start immediate code
```

```
                        ;
                        ; Internal BASIC entry points that are likely to need to be vectored
                        ; because routines unlikely to be vectored call them.
                        ;
0B70  1830                      MAC     JSB     BAS     BAS$$CB_POP
0B78  1831                      MAC     JSB     BAS     BAS$$CB_PUSH
0B80  1832                      MAC     JSB     BAS     BAS$$CB_GET
0B88  1833                      MAC     CALL    BAS     BAS$$ERR_INIT
0B90  1834                      MAC     CALL    BAS     BAS$$OPEN_ZERO
0B98  1835                      MAC     CALL    BAS     BAS$$RECOD_INIT
0BA0  1836                      MAC     CALL    BAS     BAS$$BLNK_LINE
0BA8  1837                      MAC     CALL    BAS     BAS$$SIGNAL
0BB0  1838                      MAC     CALL    BAS     BAS$$SIGNAL_IO
0BB8  1839                      MAC     CALL    BAS     BAS$$STATU_INIT
0BC0  1840                      MAC     CALL    BAS     BAS$$STOP
0BC8  1841                      MAC     CALL    BAS     BAS$$STOP_IO
0BD0  1842                      MAC     CALL    BAS     BAS$CANTYPAHEAD
0BD8  1843                      MAC     JSB     BAS     BAS$$SCALE_L_R1
0BE0  1844                      MAC     JSB     BAS     BAS$$SCALE_RT
0BE8  1845                      MAC     CALL    BAS     BAS$$STOP_RMS
0BF0  1846                      MAC     CALL    BAS     BAS$$FORMAT_INT
0BF8  1847                      MAC     CALL    BAS     BAS$$CLOSE_ALL
0C00  1848                      MAC     CALL    BAS     BAS$$UDF_RL1
0C08  1849                      MAC     CALL    BAS     BAS$$UDF_WL1
0C10  1850                      ;
0C10  1851                      ; JSB entry points to the string routines.
0C10  1852                      ;
0C10  1853                      MAC     JSB     STR     STR$COPY_DX_R8
0C18  1854                      MAC     JSB     STR     STR$COPY_R_R8
0C20  1855                      MAC     JSB     STR     STR$DUPL_CHARR8
0C28  1856                      MAC     JSB     STR     STR$FREE1_DX_R4
0C30  1857                      MAC     JSB     STR     STR$GET1_DX_R4
0C38  1858                      MAC     JSB     STR     STR$LEFT_R8
0C40  1859                      MAC     JSB     STR     STR$LEN_EXTR_R8
0C48  1860                      MAC     JSB     STR     STR$POSITION_R6
0C50  1861                      MAC     JSB     STR     STR$POS_EXTR_R8
0C58  1862                      MAC     JSB     STR     STR$RIGHT_R8
0C60  1863                      ;
0C60  1864                      ; More STR$ entry points.  These modules must be in the sharable
0C60  1865                      ; library, even though they are not used much, because they use
0C60  1866                      ; string interlocks.
0C60  1867                      ;
0C60  1868                      MAC     CALL    STR     STR$APPEND
0C68  1869                      MAC     CALL    STR     STR$COMPARE
0C70  1870                      MAC     CALL    STR     STR$COMPARE_EQL
0C78  1871                      MAC     CALL    STR     STR$PREFIX
0C80  1872                      MAC     CALL    STR     STR$REPLACE
0C88  1873                      MAC     JSB     STR     STR$REPLACE_R8
0C90  1874                      MAC     CALL    STR     STR$TRANSLATE
0C98  1875                      MAC     CALL    STR     STR$UPCASE
```

```
                        ;
                        ; The BASIC error codes.  First the small integer symbols.
                        ;
                                MAC     SYM     BAS     BAS$K_ACCDEVUSE
                                MAC     SYM     BAS     BAS$K_ARGDONMAT
                                MAC     SYM     BAS     BAS$K_ARGOUTBOU
                                MAC     SYM     BAS     BAS$K_ARGTOOLAR
                                MAC     SYM     BAS     BAS$K_ARRMUSSAM
                                MAC     SYM     BAS     BAS$K_ARRMUSSQU
                                MAC     SYM     BAS     BAS$K_BADDIRDEV
                                MAC     SYM     BAS     BAS$K_BADLINNUM
                                MAC     SYM     BAS     BAS$K_BADNUMPRI
                                MAC     SYM     BAS     BAS$K_BADRECIDE
                                MAC     SYM     BAS     BAS$K_BADRECVAL
                                MAC     SYM     BAS     BAS$K_CANCHAARR
                                MAC     SYM     BAS     BAS$K_CANCOMSTA
                                MAC     SYM     BAS     BAS$K_CANCON
                                MAC     SYM     BAS     BAS$K_CANFINFIL
                                MAC     SYM     BAS     BAS$K_CANINVMAT
                                MAC     SYM     BAS     BAS$K_CANOPEFIL
                                MAC     SYM     BAS     BAS$K_CANPOSEOF
                                MAC     SYM     BAS     BAS$K_CHATO_NON
                                MAC     SYM     BAS     BAS$K_CORFICSTR
                                MAC     SYM     BAS     BAS$K_DATFORERR
                                MAC     SYM     BAS     BAS$K_DATTYPERR
                                MAC     SYM     BAS     BAS$K_DEFWITFNE
                                MAC     SYM     BAS     BAS$K_DEVHUNWRI
                                MAC     SYM     BAS     BAS$K_DEVNOTAVA
                                MAC     SYM     BAS     BAS$K_DEVNOTFIL
                                MAC     SYM     BAS     BAS$K_DIFUSELON
                                MAC     SYM     BAS     BAS$K_DIRERR
                                MAC     SYM     BAS     BAS$K_DISBLOINT
                                MAC     SYM     BAS     BAS$K_DISERRDUR
                                MAC     SYM     BAS     BAS$K_DISPACLOC
                                MAC     SYM     BAS     BAS$K_DISPACNEE
                                MAC     SYM     BAS     BAS$K_DISPACNOT
                                MAC     SYM     BAS     BAS$K_DISPACPRI
                                MAC     SYM     BAS     BAS$K_DIVBY_ZER
                                MAC     SYM     BAS     BAS$K_DUPKEYDET
                                MAC     SYM     BAS     BAS$K_ENDFILDEV
                                MAC     SYM     BAS     BAS$K_ERRTRANEE
                                MAC     SYM     BAS     BAS$K_ENDOF_STA
                                MAC     SYM     BAS     BAS$K_EXEONCFIL
                                MAC     SYM     BAS     BAS$K_EXPERR
                                MAC     SYM     BAS     BAS$K_EXPTOOCOM
                                MAC     SYM     BAS     BAS$K_FATDISPAC
                                MAC     SYM     BAS     BAS$K_FATSYSIO
                                MAC     SYM     BAS     BAS$K_FIEOVEBUF
                                MAC     SYM     BAS     BAS$K_FILACPFAI
                                MAC     SYM     BAS     BAS$K_FILATTNOT
                                MAC     SYM     BAS     BAS$K_FILEXIREN
                                MAC     SYM     BAS     BAS$K_FILEXPDAT
                                MAC     SYM     BAS     BAS$K_FILIS_LOC
                                MAC     SYM     BAS     BAS$K_FIRARGSEQ
                                MAC     SYM     BAS     BAS$K_FLOOVE
                                MAC     SYM     BAS     BAS$K_FLOPOIERR
                                MAC     SYM     BAS     BAS$K_FLOUND
```

```
0CA0  1934        MAC     SYM     BAS     BAS$K_FNEWITDEF
0CA0  1935        MAC     SYM     BAS     BAS$K_FNEWITFUN
0CA0  1936        MAC     SYM     BAS     BAS$K_FORWITNEX
0CA0  1937        MAC     SYM     BAS     BAS$K_ILLALLCLA
0CA0  1938        MAC     SYM     BAS     BAS$K_ILLARGLOG
0CA0  1939        MAC     SYM     BAS     BAS$K_ILLBYTCOU
0CA0  1940        MAC     SYM     BAS     BAS$K_ILLCLUSIZ
0CA0  1941        MAC     SYM     BAS     BAS$K_ILLCONCLA
0CA0  1942        MAC     SYM     BAS     BAS$K_ILLDEFNES
0CA0  1943        MAC     SYM     BAS     BAS$K_ILLDUMVAR
0CA0  1944        MAC     SYM     BAS     BAS$K_ILLEXIDEF
0CA0  1945        MAC     SYM     BAS     BAS$K_ILLEXP
0CA0  1946        MAC     SYM     BAS     BAS$K_ILLFIEVAR
0CA0  1947        MAC     SYM     BAS     BAS$K_ILLFILNAM
0CA0  1948        MAC     SYM     BAS     BAS$K_ILLFN_RED
0CA0  1949        MAC     SYM     BAS     BAS$K_ILLFUNNAM
0CA0  1950        MAC     SYM     BAS     BAS$K_ILLIF_STA
0CA0  1951        MAC     SYM     BAS     BAS$K_ILLILLACC
0CA0  1952        MAC     SYM     BAS     BAS$K_ILLIN_IMM
0CA0  1953        MAC     SYM     BAS     BAS$K_ILLIO_CHA
0CA0  1954        MAC     SYM     BAS     BAS$K_ILLKEYATT
0CA0  1955        MAC     SYM     BAS     BAS$K_ILLLINNUM
0CA0  1956        MAC     SYM     BAS     BAS$K_ILLMAGUSA
0CA0  1957        MAC     SYM     BAS     BAS$K_ILLMODMIX
0CA0  1958        MAC     SYM     BAS     BAS$K_ILLNUM
0CA0  1959        MAC     SYM     BAS     BAS$K_ILLNUMIMA
0CA0  1960        MAC     SYM     BAS     BAS$K_ILLOPE
0CA0  1961        MAC     SYM     BAS     BAS$K_ILLRECACC
0CA0  1962        MAC     SYM     BAS     BAS$K_ILLRECFIL
0CA0  1963        MAC     SYM     BAS     BAS$K_ILLRECFOR
0CA0  1964        MAC     SYM     BAS     BAS$K_ILLRESSUB
0CA0  1965        MAC     SYM     BAS     BAS$K_ILLRETSUB
0CA0  1966        MAC     SYM     BAS     BAS$K_ILLSTA
0CA0  1967        MAC     SYM     BAS     BAS$K_ILLSTRIMA
0CA0  1968        MAC     SYM     BAS     BAS$K_ILLSWIUSA
0CA0  1969        MAC     SYM     BAS     BAS$K_ILLSYM
0CA0  1970        MAC     SYM     BAS     BAS$K_ILLSYSUSA
0CA0  1971        MAC     SYM     BAS     BAS$K_ILLUSA
0CA0  1972        MAC     SYM     BAS     BAS$K_ILLUSADEV
0CA0  1973        MAC     SYM     BAS     BAS$K_ILLVER
0CA0  1974        MAC     SYM     BAS     BAS$K_IMASQUROO
0CA0  1975        MAC     SYM     BAS     BAS$K_INCFUNUSA
0CA0  1976        MAC     SYM     BAS     BAS$K_INCSUBUSE
0CA0  1977        MAC     SYM     BAS     BAS$K_INDNOTFUL
0CA0  1978        MAC     SYM     BAS     BAS$K_INDNOTINI
0CA0  1979        MAC     SYM     BAS     BAS$K_INTERR
0CA0  1980        MAC     SYM     BAS     BAS$K_INTOVEFOR
0CA0  1981        MAC     SYM     BAS     BAS$K_INVFILOPT
0CA0  1982        MAC     SYM     BAS     BAS$K_INVKEYREF
0CA0  1983        MAC     SYM     BAS     BAS$K_INVRFAFIE
0CA0  1984        MAC     SYM     BAS     BAS$K_IO_CHAALR
0CA0  1985        MAC     SYM     BAS     BAS$K_IO_CHANOT
0CA0  1986        MAC     SYM     BAS     BAS$K_IO_TO_DET
0CA0  1987        MAC     SYM     BAS     BAS$K_KEYFIEBEY
0CA0  1988        MAC     SYM     BAS     BAS$K_KEYLARTHA
0CA0  1989        MAC     SYM     BAS     BAS$K_KEYNOTCHA
0CA0  1990        MAC     SYM     BAS     BAS$K_KEYSIZTOO
```

```
                0CA0  1991          MAC     SYM     BAS     BAS$K_KEYWAIEXH
                0CA0  1992          MAC     SYM     BAS     BAS$K_LINTOOLON
                0CA0  1993          MAC     SYM     BAS     BAS$K_LITSTRNEE
                0CA0  1994          MAC     SYM     BAS     BAS$K_MAGRECLEN
                0CA0  1995          MAC     SYM     BAS     BAS$K_MAGSELERR
                0CA0  1996          MAC     SYM     BAS     BAS$K_MATARRTOO
                0CA0  1997          MAC     SYM     BAS     BAS$K_MATARRWIT
                0CA0  1998          MAC     SYM     BAS     BAS$K_MATDIMERR
                0CA0  1999          MAC     SYM     BAS     BAS$K_MAXMEMEXC
                0CA0  2000          MAC     SYM     BAS     BAS$K_MEMMANVIO
                0CA0  2001          MAC     SYM     BAS     BAS$K_MEMPARFAI
                0CA0  2002          MAC     SYM     BAS     BAS$K_MISSPEFEA
                0CA0  2003          MAC     SYM     BAS     BAS$K_MODERR
                0CA0  2004          MAC     SYM     BAS     BAS$K_MOVOVEBUF
                0CA0  2005          MAC     SYM     BAS     BAS$K_NAMACCNOW
                0CA0  2006          MAC     SYM     BAS     BAS$K_NEGFILSTR
                0CA0  2007          MAC     SYM     BAS     BAS$K_NEXWITFOR
                0CA0  2008          MAC     SYM     BAS     BAS$K_NODNAMERR
                0CA0  2009          MAC     SYM     BAS     BAS$K_NONRESRUN
                0CA0  2010          MAC     SYM     BAS     BAS$K_NOTENDFIL
                0CA0  2011          MAC     SYM     BAS     BAS$K_NOTENOAVA
                0CA0  2012          MAC     SYM     BAS     BAS$K_NOTENODAT
                0CA0  2013          MAC     SYM     BAS     BAS$K_NOTIMP
                0CA0  2014          MAC     SYM     BAS     BAS$K_NOTRANACC
                0CA0  2015          MAC     SYM     BAS     BAS$K_NOTVALDEV
                0CA0  2016          MAC     SYM     BAS     BAS$K_NO_BUFSPA
                0CA0  2017          MAC     SYM     BAS     BAS$K_NO_CURREC
                0CA0  2018          MAC     SYM     BAS     BAS$K_NO_FIEIMA
                0CA0  2019          MAC     SYM     BAS     BAS$K_NO_FILNAM
                0CA0  2020          MAC     SYM     BAS     BAS$K_NO_PRIKEY
                0CA0  2021          MAC     SYM     BAS     BAS$K_NO_ROOUSE
                0CA0  2022          MAC     SYM     BAS     BAS$K_NO_RUNSYS
                0CA0  2023          MAC     SYM     BAS     BAS$K_NUEIMA
                0CA0  2024          MAC     SYM     BAS     BAS$K_NUMIMASTR
                0CA0  2025          MAC     SYM     BAS     BAS$K_NUMIS_NEE
                0CA0  2026          MAC     SYM     BAS     BAS$K_ODDADDTRA
                0CA0  2027          MAC     SYM     BAS     BAS$K_ONEOR_TWO
                0CA0  2028          MAC     SYM     BAS     BAS$K_ON_STANEE
                0CA0  2029          MAC     SYM     BAS     BAS$K_ON_STAOUT
                0CA0  2030          MAC     SYM     BAS     BAS$K_OUTOF_DAT
                0CA0  2031          MAC     SYM     BAS     BAS$K_PACIDSDON
                0CA0  2032          MAC     SYM     BAS     BAS$K_PLEUSERUN
                0CA0  2033          MAC     SYM     BAS     BAS$K_PRIKEYOUT
                0CA0  2034          MAC     SYM     BAS     BAS$K_PRIUSIBUF
                0CA0  2035          MAC     SYM     BAS     BAS$K_PRIUSIFOR
                0CA0  2036          MAC     SYM     BAS     BAS$K_PROC__TRA
                0CA0  2037          MAC     SYM     BAS     BAS$K_PROLOSSOR
                0CA0  2038          MAC     SYM     BAS     BAS$K_PROVIO
                0CA0  2039          MAC     SYM     BAS     BAS$K_RECALREXI
                0CA0  2040          MAC     SYM     BAS     BAS$K_RECATTNOT
                0CA0  2041          MAC     SYM     BAS     BAS$K_RECBUCLOC
                0CA0  2042          MAC     SYM     BAS     BAS$K_RECFILTOO
                0CA0  2043          MAC     SYM     BAS     BAS$K_RECHASBEE
                0CA0  2044          MAC     SYM     BAS     BAS$K_RECLOCFAI
                0CA0  2045          MAC     SYM     BAS     BAS$K_RECNOTFOU
                0CA0  2046          MAC     SYM     BAS     BAS$K_RECNUMEXC
                0CA0  2047          MAC     SYM     BAS     BAS$K_RECSUBCAL
```

```
0CA0  2048          MAC     SYM     BAS     BAS$K_REDARR
0CA0  2049          MAC     SYM     BAS     BAS$K_RESINSTRA
0CA0  2050          MAC     SYM     BAS     BAS$K_RESNO_ERR
0CA0  2051          MAC     SYM     BAS     BAS$K_RETWITGOS
0CA0  2052          MAC     SYM     BAS     BAS$K_RRVNOTFUL
0CA0  2053          MAC     SYM     BAS     BAS$K_SCAFACINT
0CA0  2054          MAC     SYM     BAS     BAS$K_SIZRECINV
0CA0  2055          MAC     SYM     BAS     BAS$K_SP_STAOVE
0CA0  2056          MAC     SYM     BAS     BAS$K_STANOTFOU
0CA0  2057          MAC     SYM     BAS     BAS$K_STO
0CA0  2058          MAC     SYM     BAS     BAS$K_STRIMANUM
0CA0  2059          MAC     SYM     BAS     BAS$K_STRIS_NEE
0CA0  2060          MAC     SYM     BAS     BAS$K_STRTOOLON
0CA0  2061          MAC     SYM     BAS     BAS$K_SUBOUTRAN
0CA0  2062          MAC     SYM     BAS     BAS$K_SYNERR
0CA0  2063          MAC     SYM     BAS     BAS$K_TAPBOTDET
0CA0  2064          MAC     SYM     BAS     BAS$K_TAPNOTANS
0CA0  2065          MAC     SYM     BAS     BAS$K_TAPRECNOT
0CA0  2066          MAC     SYM     BAS     BAS$K_TERFORFIL
0CA0  2067          MAC     SYM     BAS     BAS$K_TIMLIMEXC
0CA0  2068          MAC     SYM     BAS     BAS$K_TOOFEWARG
0CA0  2069          MAC     SYM     BAS     BAS$K_TOOMANARG
0CA0  2070          MAC     SYM     BAS     BAS$K_TOOMANOPE
0CA0  2071          MAC     SYM     BAS     BAS$K_UNDFUNCAL
0CA0  2072          MAC     SYM     BAS     BAS$K_USEDATERR
0CA0  2073          MAC     SYM     BAS     BAS$K_VIRARRDIS
0CA0  2074          MAC     SYM     BAS     BAS$K_VIRARROPE
0CA0  2075          MAC     SYM     BAS     BAS$K_VIRBUFTOO
0CA0  2076          MAC     SYM     BAS     BAS$K_WHA
0CA0  2077          MAC     SYM     BAS     BAS$K_WROMATPAC
0CA0  2078 ;
0CA0  2079 ; New messages for Basic 2.0, VMS 3.1
0CA0  2080 ;
0CA0  2081          MAC     SYM     BAS     BAS$K_NEGZERTAB
0CA0  2082          MAC     SYM     BAS     BAS$K_TOOMUCDAT
0CA0  2083          MAC     SYM     BAS     BAS$K_ERRFILCOR
0CA0  2084          MAC     SYM     BAS     BAS$K_UNEFILDAT
0CA0  2085          MAC     SYM     BAS     BAS$K_NOSUPFOR
0CA0  2086          MAC     SYM     BAS     BAS$K_DECERR
0CA0  2087          MAC     SYM     BAS     BAS$K_NETOPEREJ
0CA0  2088          MAC     SYM     BAS     BAS$K_REMOVEBUF
0CA0  2089          MAC     SYM     BAS     BAS$K_UNAREMVAR
0CA0  2090          MAC     SYM     BAS     BAS$K_RECOVEMAP
0CA0  2091          MAC     SYM     BAS     BAS$K_IMPERRHAN
0CA0  2092          MAC     SYM     BAS     BAS$K_ILLRECLOC
0CA0  2093          MAC     SYM     BAS     BAS$K_REQRECSIZ
0CA0  2094          MAC     SYM     BAS     BAS$K_TOOLITDAT
0CA0  2095
0CA0  2096 ;
0CA0  2097 ; Now the 32-bit values.
0CA0  2098 ;
0CA0  2099          MAC     SYM     BAS     BAS$_ACCDEVUSE
0CA0  2100          MAC     SYM     BAS     BAS$_ARGDONMAT
0CA0  2101          MAC     SYM     BAS     BAS$_ARGOUTBOU
0CA0  2102          MAC     SYM     BAS     BAS$_ARGTOOLAR
0CA0  2103          MAC     SYM     BAS     BAS$_ARRMUSSAM
0CA0  2104          MAC     SYM     BAS     BAS$_ARRMUSSQU
```

```
0CA0  2105          MAC     SYM     BAS     BAS$_BADDIRDEV
0CA0  2106          MAC     SYM     BAS     BAS$_BADLINNUM
0CA0  2107          MAC     SYM     BAS     BAS$_BADNUMPRI
0CA0  2108          MAC     SYM     BAS     BAS$_BADRECIDE
0CA0  2109          MAC     SYM     BAS     BAS$_BADRECVAL
0CA0  2110          MAC     SYM     BAS     BAS$_CANCHAARR
0CA0  2111          MAC     SYM     BAS     BAS$_CANCOMSTA
0CA0  2112          MAC     SYM     BAS     BAS$_CANCON
0CA0  2113          MAC     SYM     BAS     BAS$_CANFINFIL
0CA0  2114          MAC     SYM     BAS     BAS$_CANINVMAT
0CA0  2115          MAC     SYM     BAS     BAS$_CANOPEFIL
0CA0  2116          MAC     SYM     BAS     BAS$_CANPOSEOF
0CA0  2117          MAC     SYM     BAS     BAS$_CHATO_NON
0CA0  2118          MAC     SYM     BAS     BAS$_CORFICSTR
0CA0  2119          MAC     SYM     BAS     BAS$_DATFORERR
0CA0  2120          MAC     SYM     BAS     BAS$_DATTYPERR
0CA0  2121          MAC     SYM     BAS     BAS$_DEFWITFNE
0CA0  2122          MAC     SYM     BAS     BAS$_DEVHUNWRI
0CA0  2123          MAC     SYM     BAS     BAS$_DEVNOTAVA
0CA0  2124          MAC     SYM     BAS     BAS$_DEVNOTFIL
0CA0  2125          MAC     SYM     BAS     BAS$_DIFUSELON
0CA0  2126          MAC     SYM     BAS     BAS$_DIRERR
0CA0  2127          MAC     SYM     BAS     BAS$_DISBLOINT
0CA0  2128          MAC     SYM     BAS     BAS$_DISERRDUR
0CA0  2129          MAC     SYM     BAS     BAS$_DISPACLOC
0CA0  2130          MAC     SYM     BAS     BAS$_DISPACNEE
0CA0  2131          MAC     SYM     BAS     BAS$_DISPACNOT
0CA0  2132          MAC     SYM     BAS     BAS$_DISPACPRI
0CA0  2133          MAC     SYM     BAS     BAS$_DIVBY_ZER
0CA0  2134          MAC     SYM     BAS     BAS$_DUPKEYDET
0CA0  2135          MAC     SYM     BAS     BAS$_ENDFILDEV
0CA0  2136          MAC     SYM     BAS     BAS$_ENDOF_STA
0CA0  2137          MAC     SYM     BAS     BAS$_ERRTRANEE
0CA0  2138          MAC     SYM     BAS     BAS$_EXEONLFIL
0CA0  2139          MAC     SYM     BAS     BAS$_EXPERR
0CA0  2140          MAC     SYM     BAS     BAS$_EXPTOOCOM
0CA0  2141          MAC     SYM     BAS     BAS$_FATDISPAC
0CA0  2142          MAC     SYM     BAS     BAS$_FATSYSIO_
0CA0  2143          MAC     SYM     BAS     BAS$_FIEOVEBUF
0CA0  2144          MAC     SYM     BAS     BAS$_FILACPFAI
0CA0  2145          MAC     SYM     BAS     BAS$_FILATTNOT
0CA0  2146          MAC     SYM     BAS     BAS$_FILEXIREN
0CA0  2147          MAC     SYM     BAS     BAS$_FILEXPDAT
0CA0  2148          MAC     SYM     BAS     BAS$_FILIS_LOC
0CA0  2149          MAC     SYM     BAS     BAS$_FIRARGSEQ
0CA0  2150          MAC     SYM     BAS     BAS$_FLOOVE
0CA0  2151          MAC     SYM     BAS     BAS$_FLOPOIERR
0CA0  2152          MAC     SYM     BAS     BAS$_FLOUND
0CA0  2153          MAC     SYM     BAS     BAS$_FNEWITDEF
0CA0  2154          MAC     SYM     BAS     BAS$_FNEWITFUN
0CA0  2155          MAC     SYM     BAS     BAS$_FORWITNEX
0CA0  2156          MAC     SYM     BAS     BAS$_ILLALLCLA
0CA0  2157          MAC     SYM     BAS     BAS$_ILLARGLOG
0CA0  2158          MAC     SYM     BAS     BAS$_ILLBYTCOU
0CA0  2159          MAC     SYM     BAS     BAS$_ILLCLUSIZ
0CA0  2160          MAC     SYM     BAS     BAS$_ILLCONCLA
0CA0  2161          MAC     SYM     BAS     BAS$_ILLDEFNES
```

```
                0CA0  2162        MAC     SYM     BAS     BAS$_ILLDUMVAR
                0CA0  2163        MAC     SYM     BAS     BAS$_ILLEXIDEF
                0CA0  2164        MAC     SYM     BAS     BAS$_ILLEXP
                0CA0  2165        MAC     SYM     BAS     BAS$_ILLFIEVAR
                0CA0  2166        MAC     SYM     BAS     BAS$_ILLFILNAM
                0CA0  2167        MAC     SYM     BAS     BAS$_ILLFN_RED
                0CA0  2168        MAC     SYM     BAS     BAS$_ILLFUNNAM
                0CA0  2169        MAC     SYM     BAS     BAS$_ILLIF_STA
                0CA0  2170        MAC     SYM     BAS     BAS$_ILLILACC
                0CA0  2171        MAC     SYM     BAS     BAS$_ILLIN_IMM
                0CA0  2172        MAC     SYM     BAS     BAS$_ILLIO_CHA
                0CA0  2173        MAC     SYM     BAS     BAS$_ILLKEYATT
                0CA0  2174        MAC     SYM     BAS     BAS$_ILLLINNUM
                0CA0  2175        MAC     SYM     BAS     BAS$_ILLMAGUSA
                0CA0  2176        MAC     SYM     BAS     BAS$_ILLMODMIX
                0CA0  2177        MAC     SYM     BAS     BAS$_ILLNUM
                0CA0  2178        MAC     SYM     BAS     BAS$_ILLNUMIMA
                0CA0  2179        MAC     SYM     BAS     BAS$_ILLOPE
                0CA0  2180        MAC     SYM     BAS     BAS$_ILLRECACC
                0CA0  2181        MAC     SYM     BAS     BAS$_ILLRECFIL
                0CA0  2182        MAC     SYM     BAS     BAS$_ILLRECFOR
                0CA0  2183        MAC     SYM     BAS     BAS$_ILLRESSUB
                0CA0  2184        MAC     SYM     BAS     BAS$_ILLRETSUB
                0CA0  2185        MAC     SYM     BAS     BAS$_ILLSTA
                0CA0  2186        MAC     SYM     BAS     BAS$_ILLSTRIMA
                0CA0  2187        MAC     SYM     BAS     BAS$_ILLSWIUSA
                0CA0  2188        MAC     SYM     BAS     BAS$_ILLSYM
                0CA0  2189        MAC     SYM     BAS     BAS$_ILLSYSUSA
                0CA0  2190        MAC     SYM     BAS     BAS$_ILLUSA
                0CA0  2191        MAC     SYM     BAS     BAS$_ILLUSADEV
                0CA0  2192        MAC     SYM     BAS     BAS$_ILLVER
                0CA0  2193        MAC     SYM     BAS     BAS$_IMASQUROO
                0CA0  2194        MAC     SYM     BAS     BAS$_INCFUNUSA
                0CA0  2195        MAC     SYM     BAS     BAS$_INCSUBUSE
                0CA0  2196        MAC     SYM     BAS     BAS$_INDNOTFUL
                0CA0  2197        MAC     SYM     BAS     BAS$_INDNOTINI
                0CA0  2198        MAC     SYM     BAS     BAS$_INTERR
                0CA0  2199        MAC     SYM     BAS     BAS$_INTOVEFOR
                0CA0  2200        MAC     SYM     BAS     BAS$_INVFILOPT
                0CA0  2201        MAC     SYM     BAS     BAS$_INVKEYREF
                0CA0  2202        MAC     SYM     BAS     BAS$_INVRFAFIE
                0CA0  2203        MAC     SYM     BAS     BAS$_IO_CHAALR
                0CA0  2204        MAC     SYM     BAS     BAS$_IO_CHANOT
                0CA0  2205        MAC     SYM     BAS     BAS$_IO_TO_DET
                0CA0  2206        MAC     SYM     BAS     BAS$_KEYFIEBEY
                0CA0  2207        MAC     SYM     BAS     BAS$_KEYLARTHA
                0CA0  2208        MAC     SYM     BAS     BAS$_KEYNOTCHA
                0CA0  2209        MAC     SYM     BAS     BAS$_KEYSIZTOO
                0CA0  2210        MAC     SYM     BAS     BAS$_KEYWAIEXH
                0CA0  2211        MAC     SYM     BAS     BAS$_LINTOOLON
                0CA0  2212        MAC     SYM     BAS     BAS$_LITSTRNEE
                0CA0  2213        MAC     SYM     BAS     BAS$_MAGRECLEN
                0CA0  2214        MAC     SYM     BAS     BAS$_MAGSELERR
                0CA0  2215        MAC     SYM     BAS     BAS$_MATARRTOO
                0CA0  2216        MAC     SYM     BAS     BAS$_MATARRWIT
                0CA0  2217        MAC     SYM     BAS     BAS$_MATDIMERR
                0CA0  2218        MAC     SYM     BAS     BAS$_MAXMEMEXC
```

```
0CA0  2219  MAC  SYM  BAS  BAS$_MEMMANVIO
0CA0  2220  MAC  SYM  BAS  BAS$_MEMPARFAI
0CA0  2221  MAC  SYM  BAS  BAS$_MISSPEFEA
0CA0  2222  MAC  SYM  BAS  BAS$_MODERR
0CA0  2223  MAC  SYM  BAS  BAS$_MOVOVEBUF
0CA0  2224  MAC  SYM  BAS  BAS$_NAMACCNOW
0CA0  2225  MAC  SYM  BAS  BAS$_NEGFILSTR
0CA0  2226  MAC  SYM  BAS  BAS$_NEXWITFOR
0CA0  2227  MAC  SYM  BAS  BAS$_NODNAMERR
0CA0  2228  MAC  SYM  BAS  BAS$_NONRESRUN
0CA0  2229  MAC  SYM  BAS  BAS$_NOTENDFIL
0CA0  223C  MAC  SYM  BAS  BAS$_NOTENOAVA
0CA0  2231  MAC  SYM  BAS  BAS$_NOTENODAT
0CA0  2232  MAC  SYM  BAS  BAS$_NOTIMP
0CA0  2233  MAC  SYM  BAS  BAS$_NOTRANACC
0CA0  2234  MAC  SYM  BAS  BAS$_NOTVALDEV
0CA0  2235  MAC  SYM  BAS  BAS$_NO_BUFSPA
0CA0  2236  MAC  SYM  BAS  BAS$_NO_CURREC
0CA0  2237  MAC  SYM  BAS  BAS$_NO_FIEIMA
0CA0  2238  MAC  SYM  BAS  BAS$_NO_FILNAM
0CA0  2239  MAC  SYM  BAS  BAS$_NO_PRIKEY
0CA0  2240  MAC  SYM  BAS  BAS$_NO_ROOUSE
0CA0  2241  MAC  SYM  BAS  BAS$_NO_RUNSYS
0CA0  2242  MAC  SYM  BAS  BAS$_NUMIMA
0CA0  2243  MAC  SYM  BAS  BAS$_NUMIMASTR
0CA0  2244  MAC  SYM  BAS  BAS$_NUMIS_NEE
0CA0  2245  MAC  SYM  BAS  BAS$_ODDADDTRA
0CA0  2246  MAC  SYM  BAS  BAS$_ONEOR_TWO
0CA0  2247  MAC  SYM  BAS  BAS$_ON_STANEE
0CA0  2248  MAC  SYM  BAS  BAS$_ON_STAOUT
0CA0  2249  MAC  SYM  BAS  BAS$_OUTOF_DAT
0CA0  2250  MAC  SYM  BAS  BAS$_PACIDSDON
0CA0  2251  MAC  SYM  BAS  BAS$_PLEUSERUN
0CA0  2252  MAC  SYM  BAS  BAS$_PRIKEYOUT
0CA0  2253  MAC  SYM  BAS  BAS$_PRIUSIBUF
0CA0  2254  MAC  SYM  BAS  BAS$_PRIUSIFOR
0CA0  2255  MAC  SYM  BAS  BAS$_PROC_TRA
0CA0  2256  MAC  SYM  BAS  BAS$_PROLOSSOR
0CA0  2257  MAC  SYM  BAS  BAS$_PROVIO
0CA0  2258  MAC  SYM  BAS  BAS$_RECALREXI
0CA0  2259  MAC  SYM  BAS  BAS$_RECATTNOT
0CA0  2260  MAC  SYM  BAS  BAS$_RECBUCLOC
0CA0  2261  MAC  SYM  BAS  BAS$_RECFILTOO
0CA0  2262  MAC  SYM  BAS  BAS$_RECHASBEE
0CA0  2263  MAC  SYM  BAS  BAS$_RECLOCFAI
0CA0  2264  MAC  SYM  BAS  BAS$_RECNOTFOU
0CA0  2265  MAC  SYM  BAS  BAS$_RECNUMEXC
0CA0  2266  MAC  SYM  BAS  BAS$_RECSUBCAL
0CA0  2267  MAC  SYM  BAS  BAS$_REDARR
0CA0  2268  MAC  SYM  BAS  BAS$_RESINSTRA
0CA0  2269  MAC  SYM  BAS  BAS$_RESNO_ERR
0CA0  2270  MAC  SYM  BAS  BAS$_RETWITGOS
0CA0  2271  MAC  SYM  BAS  BAS$_RRVNOTFUL
0CA0  2272  MAC  SYM  BAS  BAS$_SCAFACINT
0CA0  2273  MAC  SYM  BAS  BAS$_SIZRECINV
0CA0  2274  MAC  SYM  BAS  BAS$_SP_STAOVE
0CA0  2275  MAC  SYM  BAS  BAS$_STANOTFOU
```

```
0CA0  2276          MAC     SYM     BAS     BAS$_STO
0CA0  2277          MAC     SYM     BAS     BAS$_STRIMANUM
0CA0  2278          MAC     SYM     BAS     BAS$_STRIS_NEE
0CA0  2279          MAC     SYM     BAS     BAS$_STRTOOLON
0CA0  2280          MAC     SYM     BAS     BAS$_SUBOUTRAN
0CA0  2281          MAC     SYM     BAS     BAS$_SYNERR
0CA0  2282          MAC     SYM     BAS     BAS$_TAPBOTDET
0CA0  2283          MAC     SYM     BAS     BAS$_TAPNOTANS
0CA0  2284          MAC     SYM     BAS     BAS$_TAPRECNOT
0CA0  2285          MAC     SYM     BAS     BAS$_TERFORFIL
0CA0  2286          MAC     SYM     BAS     BAS$_TIMLIMEXC
0CA0  2287          MAC     SYM     BAS     BAS$_TOOFEWARG
0CA0  2288          MAC     SYM     BAS     BAS$_TOOMANARG
0CA0  2289          MAC     SYM     BAS     BAS$_TOOMANOPE
0CA0  2290          MAC     SYM     BAS     BAS$_UNDFUNCAL
0CA0  2291          MAC     SYM     BAS     BAS$_USEDATERR
0CA0  2292          MAC     SYM     BAS     BAS$_VIRARRDIS
0CA0  2293          MAC     SYM     BAS     BAS$_VIRARROPE
0CA0  2294          MAC     SYM     BAS     BAS$_VIRBUFTOO
0CA0  2295          MAC     SYM     BAS     BAS$_WHA
0CA0  2296          MAC     SYM     BAS     BAS$_WROMATPAC
0CA0  2297
0CA0  2298 ; New messages for Basic 2.0, VMS 3.1
0CA0  2299          MAC     SYM     BAS     BAS$_NEGZERTAB
0CA0  2300          MAC     SYM     BAS     BAS$_TOOMUCDAT
0CA0  2301          MAC     SYM     BAS     BAS$_ERRFILCOR
0CA0  2302          MAC     SYM     BAS     BAS$_UNEFILDAT
0CA0  2303          MAC     SYM     BAS     BAS$_NOSUPFOR
0CA0  2304          MAC     SYM     BAS     BAS$_DECERR
0CA0  2305          MAC     SYM     BAS     BAS$_NETOPEREJ
0CA0  2306          MAC     SYM     BAS     BAS$_REMOVEBUF
0CA0  2307          MAC     SYM     BAS     BAS$_UNAREMVAR
0CA0  2308          MAC     SYM     BAS     BAS$_RECOVEMAP
0CA0  2309          MAC     SYM     BAS     BAS$_IMPERRHAN
0CA0  2310          MAC     SYM     BAS     BAS$_ILLRECLOC
0CA0  2311          MAC     SYM     BAS     BAS$_REQRECSIZ
0CA0  2312          MAC     SYM     BAS     BAS$_TOOLITDAT
0CA0  2313
0CA0  2314 ; Module BAS$$REC_PROC
0CA0  2315          MAC     CALL    BAS     BASSWAIT
```

```
                                0CA8  2517 ;+
                                0CA8  2518 ; START OF COBOL CONTRIBUTION TO RTLVECTOR
                                0CA8  2519 ;
                                0CA8  2520 ; Leave some expansion room for BAS by padding to a page boundary.
                                0CA8  2521 ; If this room is exhausted they will have to go after
                                0CA8  2522 ; the COB facility.  Give an error message if the space gets
                                0CA8  2523 ; exhausted.
                                0CA8  2524 ;-
                    FFFFFEAB    0CA8  2525         .IF GREATER <<.-RTL$START>-^XE00>
                                0CA8  2526             .ERROR <<.-RTL$START>-^XE00>    ; Negative vector pad space
                                0CA8  2527         .ENDC
00'00'00'00'00'00'00'00'00'00'  0CA8  2528         .BYTE   0[512-<<.-RTL$START>&511>]
00'00'00'00'00'00'00'00'00'00'  0CB4
00'00'00'00'00'00'00'00'00'00'  0CC0
00'00'00'00'00'00'00'00'00'00'  0CCC
00'00'00'00'00'00'00'00'00'00'  0CD8
00'00'00'00'00'00'00'00'00'00'  0CE4
00'00'00'00'00'00'00'00'00'00'  0CF0
00'00'00'00'00'00'00'00'00'00'  0CFC
00'00'00'00'00'00'00'00'00'00'  0D08
00'00'00'00'00'00'00'00'00'00'  0D14
00'00'00'00'00'00'00'00'00'00'  0D20
00'00'00'00'00'00'00'00'00'00'  0D2C
00'00'00'00'00'00'00'00'00'00'  0D38
00'00'00'00'00'00'00'00'00'00'  0D44
00'00'00'00'00'00'00'00'00'00'  0D50
00'00'00'00'00'00'00'00'00'00'  0D5C
00'00'00'00'00'00'00'00'00'00'  0D68
00'00'00'00'00'00'00'00'00'00'  0D74
00'00'00'00'00'00'00'00'00'00'  0D80
00'00'00'00'00'00'00'00'00'00'  0D8C
00'00'00'00'00'00'00'00'00'00'  0D98
00'00'00'00'00'00'00'00'00'00'  0DA4
00'00'00'00'00'00'00'00'00'00'  0DB0
00'00'00'00'00'00'00'00'00'00'  0DBC
00'00'00'00'00'00'00'00'00'00'  0DC8
00'00'00'00'00'00'00'00'00'00'  0DD4
00'00'00'00'00'00'00'00'00'00'  0DE0
00'00'00'00'00'00'00'00'00'00'  0DEC
          00'00'00'00'00'00'00'00'  0DF8
```

```
0E00  2330 ; MODULE LIB$AB_CVTPT_O
0E00  2331          MAC     SYM     LIB     LIB$AB_CVTPT_O
0E00  2332
0E00  2333 ; MODULE LIB$AB_CVTPT_U
0E00  2334          MAC     SYM     LIB     LIB$AB_CVTPT_U
0E00  2335
0E00  2336 ; MODULE LIB$AB_CVTTP_O
0E00  2337          MAC     SYM     LIB     LIB$AB_CVTTP_O
0E00  2338
0E00  2339 ; MODULE LIB$AB_CVTTP_U
0E00  2340          MAC     SYM     LIB     LIB$AB_CVTTP_U
0E00  2341
0E00  2342 ; MODULE COB$AB_SPANC
0E00  2343          MAC     SYM     COB     COB$AB_SPANC
0E00  2344
0E00  2345 ; MODULE LIB$AB_CVT_U_O
0E00  2346          MAC     SYM     LIB     LIB$AB_CVT_U_O
0E00  2347
0E00  2348 ;+
0E00  2349 ; Degree equivalents of trig functions
0E00  2350 ;-
0E00  2351
0E00  2352 ; MODULE:MTH$ACOS
0E00  2353          MAC     CALL    MTH     MTH$ACOSD
0E08  2354          MAC     JSB     MTH     MTH$ACOSD_R4
0E10  2355
0E10  2356 ; MODULE:MTH$ASIN
0E10  2357          MAC     CALL    MTH     MTH$ASIND
0E18  2358          MAC     JSB     MTH     MTH$ASIND_R4
0E20  2359
0E20  2360 ; MODULE:MTH$ATAN
0E20  2361          MAC     CALL    MTH     MTH$ATAND
0E28  2362          MAC     CALL    MTH     MTH$ATAND2
0E30  2363          MAC     JSB     MTH     MTH$ATAND_R4
0E38  2364
0E38  2365 ; MODULE:MTH$DACOS
0E38  2366          MAC     CALL    MTH     MTH$DACOSD
0E40  2367          MAC     JSB     MTH     MTH$DACOSD_R7
0E48  2368
0E48  2369 ; MODULE:MTH$DASIN
0E48  2370          MAC     CALL    MTH     MTH$DASIND
0E50  2371          MAC     JSB     MTH     MTH$DASIND_R7
0E58  2372
0E58  2373 ; MODULE:MTH$DATAN
0E58  2374          MAC     CALL    MTH     MTH$DATAND
0E60  2375          MAC     CALL    MTH     MTH$DATAND2
0E68  2376          MAC     JSB     MTH     MTH$DATAND_R7
0E70  2377
0E70  2378 ; MODULE:MTH$DSINCOS
0E70  2379          MAC     CALL    MTH     MTH$DCOSD
0E78  2380          MAC     JSB     MTH     MTH$DCOSD_R7
0E80  2381          MAC     CALL    MTH     MTH$DSIND
0E88  2382          MAC     JSB     MTH     MTH$DSIND_R7
0E90  2383
0E90  2384 ; MODULE:MTH$SINCOS
0E90  2385          MAC     CALL    MTH     MTH$COSD
0E98  2386          MAC     JSB     MTH     MTH$COSD_R4
```

```
0EA0  2387          MAC     CALL    MTH     MTH$SIND
0EA8  2388          MAC     JSB     MTH     MTH$SIND_R4
0EB0  2389
0EB0  2390 ; MODULE:MTH$DTAN
0EB0  2391          MAC     CALL    MTH     MTH$DTAND
0EB8  2392          MAC     JSB     MTH     MTH$DTAND_R7
0EC0  2393
0EC0  2394 ; MODULE:MTH$TAN
0EC0  2395          MAC     CALL    MTH     MTH$TAND
0EC8  2396          MAC     JSB     MTH     MTH$TAND_R4
0ED0  2397
0ED0  2398 ; MODULE MTH$GACOS
0ED0  2399          MAC     NOVECT  MTH     MTH$GACOSD
0ED0  2400          MAC     NOVECT  MTH     MTH$GACOSD_R7
0ED0  2401
0ED0  2402 ; MODULE MTH$GASIN
0ED0  2403          MAC     NOVECT  MTH     MTH$GASIND
0ED0  2404          MAC     NOVECT  MTH     MTH$GASIND_R7
0ED0  2405
0ED0  2406 ; MODULE MTH$GSINCOS
0ED0  2407          MAC     NOVECT  MTH     MTH$GSIND
0ED0  2408          MAC     NOVECT  MTH     MTH$GCOSD
0ED0  2409          MAC     NOVECT  MTH     MTH$GSIND_R7
0ED0  2410          MAC     NOVECT  MTH     MTH$GCOSD_R7
0ED0  2411
0ED0  2412 ; MODULE MTH$GTAN
0ED0  2413          MAC     NOVECT  MTH     MTH$GTAND
0ED0  2414          MAC     NOVECT  MTH     MTH$GTAND_R7
0ED0  2415
0ED0  2416 ; MODULE MTH$HACOS
0ED0  2417          MAC     NOVECT  MTH     MTH$HACOSD
0ED0  2418          MAC     NOVECT  MTH     MTH$HACOSD_R8
0ED0  2419
0ED0  2420 ; MODULE MTH$HASIN
0ED0  2421          MAC     NOVECT  MTH     MTH$HASIND
0ED0  2422          MAC     NOVECT  MTH     MTH$HASIND_R8
0ED0  2423
0ED0  2424 ; MODULE MTH$HATANH
0ED0  2425          MAC     NOVECT  MTH     MTH$HATANH
0ED0  2426
0ED0  2427 ; MODULE MTH$HSINCOS
0ED0  2428          MAC     NOVECT  MTH     MTH$HSIND
0ED0  2429          MAC     NOVECT  MTH     MTH$HSIND_R5
0ED0  2430          MAC     NOVECT  MTH     MTH$HCOSD
0ED0  2431          MAC     NOVECT  MTH     MTH$HCOSD_R5
0ED0  2432
0ED0  2433 ; MODULE MTH$HTAN
0ED0  2434          MAC     NOVECT  MTH     MTH$HTAND
0ED0  2435          MAC     NOVECT  MTH     MTH$HTAND_R5
0ED0  2436
0ED0  2437 ; V E R S I O N   3 . 0   A D D I T I O N S
0ED0  2438 ;
0ED0  2439 ; Fill up the hole created by removing some translation tables that ended up
0ED0  2440 ; in the vector.
0ED0  2441
0ED0  2442 ; MODULE:MTH$ATANH
0ED0  2443          MAC     CALL    MTH     MTH$ATANH
```

```
0ED8  2444
0ED8  2445 ; MODULE:MTH$DATANH
0ED8  2446          MAC     CALL    MTH     MTH$DATANH
0EE0  2447
0EE0  2448 ; MODULE MTH$GATAN
0EE0  2449          MAC     NOVECT  MTH     MTH$GATAN
0EE0  2450          MAC     NOVECT  MTH     MTH$GATAN2
0EE0  2451          MAC     NOVECT  MTH     MTH$GATAN_R7
0EE0  2452          MAC     NOVECT  MTH     MTH$GATAND
0EE0  2453          MAC     NOVECT  MTH     MTH$GATAND2
0EE0  2454          MAC     NOVECT  MTH     MTH$GATAND_R7
0EE0  2455
0EE0  2456 ; MODULE MTH$GLOG
0EE0  2457          MAC     NOVECT  MTH     MTH$GLOG
0EE0  2458          MAC     NOVECT  MTH     MTH$GLOG2
0EE0  2459          MAC     NOVECT  MTH     MTH$GLOG10
0EE0  2460          MAC     NOVECT  MTH     MTH$GLOG_R8
0EE0  2461          MAC     NOVECT  MTH     MTH$GLOGT0_R8
0EE0  2462
0EE0  2463 ; MODULE MTH$HATAN
0EE0  2464          MAC     NOVECT  MTH     MTH$HATAN
0EE0  2465          MAC     NOVECT  MTH     MTH$HATAN_R8
0EE0  2466          MAC     NOVECT  MTH     MTH$HATAN2
0EE0  2467          MAC     NOVECT  MTH     MTH$HATAND
0EE0  2468          MAC     NOVECT  MTH     MTH$HATAND_R8
0EE0  2469          MAC     NOVECT  MTH     MTH$HATAND2
0EE0  2470
0EE0  2471 ; MODULE MTH$HLOG
0EE0  2472          MAC     NOVECT  MTH     MTH$HLOG
0EE0  2473          MAC     NOVECT  MTH     MTH$HLOG2
0EE0  2474          MAC     NOVECT  MTH     MTH$HLOG10
0EE0  2475          MAC     NOVECT  MTH     MTH$HLOG_R8
0EE0  2476          MAC     NOVECT  MTH     MTH$HLOGT0_R8
0EE0  2477
0EE0  2478 ; MODULE MTH$SINCOS      (Continued)
0EE0  2479          MAC     CALL    MTH     MTH$SINCOS
0EE8  2480          MAC     JSB     MTH     MTH$SINCOS_R5
0EF0  2481          MAC     CALL    MTH     MTH$SINCOSD
0EF8  2482          MAC     JSB     MTH     MTH$SINCOSD_R5
0F00  2483
0F00  2484 ; MODULE MTH$DSINCOS     (Continued)
0F00  2485          MAC     CALL    MTH     MTH$DSINCOS
0F08  2486          MAC     JSB     MTH     MTH$DSINCOS_R7
0F10  2487          MAC     CALL    MTH     MTH$DSINCOSD
0F18  2488          MAC     JSB     MTH     MTH$DSINCOSD_R7
0F20  2489
0F20  2490 ; MODULE MTH$GSINCOS     (Continued)
0F20  2491          MAC     NOVECT  MTH     MTH$GSINCOS
0F20  2492          MAC     NOVECT  MTH     MTH$GSINCOS_R7
0F20  2493          MAC     NOVECT  MTH     MTH$GSINCOSD
0F20  2494          MAC     NOVECT  MTH     MTH$GSINCOSD_R7
0F20  2495
0F20  2496 ; MODULE MTH$HSINCOS     (Continued)
0F20  2497          MAC     NOVECT  MTH     MTH$HSINCOS
0F20  2498          MAC     NOVECT  MTH     MTH$HSINCOS_R7
0F20  2499          MAC     NOVECT  MTH     MTH$HSINCOSD
0F20  2500          MAC     NOVECT  MTH     MTH$HSINCOSD_R7
```

```
0F20  2501
0F20  2502 ; MODULE:MTH$ALOG          (Continued)
0F20  2503          MAC     CALL    MTH     MTH$ALOG2
0F28  2504
0F28  2505 ; MODULE:MTH$DLOG          (Continued)
0F28  2506          MAC     CALL    MTH     MTH$DLOG2
0F30  2507
0F30  2508 ; MODULE MTH$AL_4_OV_PI
0F30  2509          MAC     NOVECT  MTH     MTH$AL_4_OV_PI
0F30  2510
0F30  2511 ; MODULE MTH$TAN           (Continued)
0F30  2512          MAC     JSB     MTH     MTH$TAN_R5
0F38  2513          MAC     JSB     MTH     MTH$TAND_R5
0F40  2514
0F40  2515 ; MODULE MTH$HTAN          (Continued)
0F40  2516          MAC     NOVECT  MTH     MTH$HTAN_R7
0F40  2517          MAC     NOVECT  MTH     MTH$HTAND_R7
0F40  2518
0F40  2519 ; MODULE MTH$AL_4_OV_PI
0F40  2520          MAC     DATA    MTH     MTH$AL_4_OV_PI
0F48  2521
0F48  2522 ; MODULE MTH$ALOG
0F48  2523          MAC     DATA    MTH     MTH$$AB_ALOG
0F50  2524
0F50  2525 ; MODULE MTH$ATAN
0F50  2526          MAC     DATA    MTH     MTH$$AB_ATAN
```

```
                                0F58  2528 ;+
                                0F58  2529 ; The previous space had been taken up by translate tables which were
                                0F58  2530 ; inadvertently non-vectored.  We decided to remove them from the
                                0F58  2531 ; vector.  Therefore, the previous space can be used for 163 new vectored
                                0F58  2532 ; subroutines.  Move the current location to be the proper spot for
                                0F58  2533 ; vectors which follow.
                                0F58  2534 ;-
                    FFFFFC40    0F58  2535         .IF GREATER <<.-RTL$START>-^X1318>
                                0F58  2536             .ERROR <<.-RTL$START>-^X1318>    ; Negative vector pad space
                                0F58  2537         .ENDC
00'00'00'00'00'00'00'00'00'00'00'00'  0F58  2538         .BYTE   0[^X1318-<.-RTL$START>]
00'00'00'00'00'00'00'00'00'00'00'00'  0F64
00'00'00'00'00'00'00'00'00'00'00'00'  0F70
00'00'00'00'00'00'00'00'00'00'00'00'  0F7C
00'00'00'00'00'00'00'00'00'00'00'00'  0F88
00'00'00'00'00'00'00'00'00'00'00'00'  0F94
00'00'00'00'00'00'00'00'00'00'00'00'  0FA0
00'00'00'00'00'00'00'00'00'00'00'00'  0FAC
00'00'00'00'00'00'00'00'00'00'00'00'  0FB8
00'00'00'00'00'00'00'00'00'00'00'00'  0FC4
00'00'00'00'00'00'00'00'00'00'00'00'  0FD0
00'00'00'00'00'00'00'00'00'00'00'00'  0FDC
00'00'00'00'00'00'00'00'00'00'00'00'  0FE8
00'00'00'00'00'00'00'00'00'00'00'00'  0FF4
00'00'00'00'00'00'00'00'00'00'00'00'  1000
00'00'00'00'00'00'00'00'00'00'00'00'  100C
00'00'00'00'00'00'00'00'00'00'00'00'  1018
00'00'00'00'00'00'00'00'00'00'00'00'  1024
00'00'00'00'00'00'00'00'00'00'00'00'  1030
00'00'00'00'00'00'00'00'00'00'00'00'  103C
00'00'00'00'00'00'00'00'00'00'00'00'  1048
00'00'00'00'00'00'00'00'00'00'00'00'  1054
00'00'00'00'00'00'00'00'00'00'00'00'  1060
00'00'00'00'00'00'00'00'00'00'00'00'  106C
00'00'00'00'00'00'00'00'00'00'00'00'  1078
00'00'00'00'00'00'00'00'00'00'00'00'  1084
00'00'00'00'00'00'00'00'00'00'00'00'  1090
00'00'00'00'00'00'00'00'00'00'00'00'  109C
00'00'00'00'00'00'00'00'00'00'00'00'  10A8
00'00'00'00'00'00'00'00'00'00'00'00'  10B4
00'00'00'00'00'00'00'00'00'00'00'00'  10C0
00'00'00'00'00'00'00'00'00'00'00'00'  10CC
00'00'00'00'00'00'00'00'00'00'00'00'  10D8
00'00'00'00'00'00'00'00'00'00'00'00'  10E4
00'00'00'00'00'00'00'00'00'00'00'00'  10F0
00'00'00'00'00'00'00'00'00'00'00'00'  10FC
00'00'00'00'00'00'00'00'00'00'00'00'  1108
00'00'00'00'00'00'00'00'00'00'00'00'  1114
00'00'00'00'00'00'00'00'00'00'00'00'  1120
00'00'00'00'00'00'00'00'00'00'00'00'  112C
00'00'00'00'00'00'00'00'00'00'00'00'  1138
00'00'00'00'00'00'00'00'00'00'00'00'  1144
00'00'00'00'00'00'00'00'00'00'00'00'  1150
00'00'00'00'00'00'00'00'00'00'00'00'  115C
00'00'00'00'00'00'00'00'00'00'00'00'  1168
00'00'00'00'00'00'00'00'00'00'00'00'  1174
00'00'00'00'00'00'00'00'00'00'00'00'  1180
```

```
00'00'00'00'00'00'00'00'00'00'00'00' 118C
00'00'00'00'00'00'00'00'00'00'00'00' 1198
00'00'00'00'00'00'00'00'00'00'00'00' 11A4
00'00'00'00'00'00'00'00'00'00'00'00' 11B0
00'00'00'00'00'00'00'00'00'00'00'00' 11BC
00'00'00'00'00'00'00'00'00'00'00'00' 11C8
00'00'00'00'00'00'00'00'00'00'00'00' 11D4
00'00'00'00'00'00'00'00'00'00'00'00' 11E0
00'00'00'00'00'00'00'00'00'00'00'00' 11EC
00'00'00'00'00'00'00'00'00'00'00'00' 11F8
00'00'00'00'00'00'00'00'00'00'00'00' 1204
00'00'00'00'00'00'00'00'00'00'00'00' 1210
00'00'00'00'00'00'00'00'00'00'00'00' 121C
00'00'00'00'00'00'00'00'00'00'00'00' 1228
00'00'00'00'00'00'00'00'00'00'00'00' 1234
00'00'00'00'00'00'00'00'00'00'00'00' 1240
00'00'00'00'00'00'00'00'00'00'00'00' 124C
00'00'00'00'00'00'00'00'00'00'00'00' 1258
00'00'00'00'00'00'00'00'00'00'00'00' 1264
00'00'00'00'00'00'00'00'00'00'00'00' 1270
00'00'00'00'00'00'00'00'00'00'00'00' 127C
00'00'00'00'00'00'00'00'00'00'00'00' 1288
00'00'00'00'00'00'00'00'00'00'00'00' 1294
00'00'00'00'00'00'00'00'00'00'00'00' 12A0
00'00'00'00'00'00'00'00'00'00'00'00' 12AC
00'00'00'00'00'00'00'00'00'00'00'00' 12B8
00'00'00'00'00'00'00'00'00'00'00'00' 12C4
00'00'00'00'00'00'00'00'00'00'00'00' 12D0
00'00'00'00'00'00'00'00'00'00'00'00' 12DC
00'00'00'00'00'00'00'00'00'00'00'00' 12E8
00'00'00'00'00'00'00'00'00'00'00'00' 12F4
00'00'00'00'00'00'00'00'00'00'00'00' 1300
00'00'00'00'00'00'00'00'00'00'00'00' 130C
                                    1318  2539
                                    1318  2540          MAC     CALL    COB     COB$HANDLER     COB$$HANDLER
```

```
                                ; MODULE COB$IOEXCEPTION -- I/O error processing
                                        MAC     CALL    COB     COB$IOEXCEPTION

                                ; MODULE COB$ERROR -- Process compiled-code-detected errors
                                        MAC     CALL    COB     COB$ERROR

                                ; MODULE COB$INTARI -- Intermediate Data Type Arithmetic
                                        MAC     CALL    COB     COB$ADDI
                                        MAC     CALL    COB     COB$SUBI
                                        MAC     CALL    COB     COB$MULI
                                        MAC     CALL    COB     COB$DIVI
                                        MAC     CALL    COB     COB$DIVI_OSE
                                        MAC     CALL    COB     COB$CMPI

                                ; MODULE COB$INTER -- Conversions to and from Intermediate Data Type
                                        MAC     JSB     COB     COB$CVTDI_R7
                                        MAC     JSB     COB     COB$CVTFI_R7
                                        MAC     JSB     COB     COB$CVTID_R7
                                        MAC     JSB     COB     COB$CVTIF_R7
                                        MAC     JSB     COB     COB$CVTIL_R8
                                        MAC     JSB     COB     COB$CVTIP_R9
                                        MAC     JSB     COB     COB$CVTIQ_R8
                                        MAC     JSB     COB     COB$CVTIW_R8
                                        MAC     JSB     COB     COB$CVTLI_R8
                                        MAC     JSB     COB     COB$CVTPI_R9
                                        MAC     JSB     COB     COB$CVTQI_R8
                                        MAC     JSB     COB     COB$CVTRIL_R8
                                        MAC     JSB     COB     COB$CVTRIP_R9
                                        MAC     JSB     COB     COB$CVTRIQ_R8
                                        MAC     JSB     COB     COB$CVTRIW_R8
                                        MAC     JSB     COB     COB$CVTTI_R8
                                        MAC     JSB     COB     COB$CVTWI_R8

                                ; MODULE COB$ACC_DATE -- Support for ACCEPT DATE
                                        MAC     CALL    COB     COB$ACC_DATE

                                ; MODULE COB$ACC_DAY -- Support for ACCEPT DAY
                                        MAC     CALL    COB     COB$ACC_DAY

                                ; MODULE COB$ACC_DAYWEEK -- ACCEPT DAY_OF_WEEK
                                        MAC     CALL    COB     COB$ACC_DAYWEEK

                                ; MODULE COB$ACC_TIME -- Support for ACCEPT TIME
                                        MAC     CALL    COB     COB$ACC_TIME

                                ; MODULE COB$ACCEPT -- Support for ACCEPT
                                        MAC     CALL    COB     COB$ACCEPT

                                ; MODULE COB$DISPLAY -- Support for DISPLAY and DISPLAY WITH NO ADVANCING
                                        MAC     CALL    COB     COB$DISPLAY
                                        MAC     CALL    COB     COB$DISP_NO_ADV

                                ; MODULE COB$DIVQ_R8 -- Quadword division
                                        MAC     JSB     COB     COB$DIVQ_R8

                                ; MODULE COB$MULQ_R8 -- Quadword multiplication
                                        MAC     JSB     COB     COB$MULQ_R8
```

```
1430  2599
1430  2600 ; MODULE COB$PAUSE -- Support for STOP
1430  2601         MAC     CALL    COB     COB$PAUSE
1438  2602
1438  2603 ; MODULE COB$CVTPQ_R9 -- Packed to Quad conversion
1438  2604         MAC     JSB     COB     COB$CVTPQ_R9
1440  2605
1440  2606 ; MODULE COB$CVTQP_R9 -- Quad to Packed conversion
1440  2607         MAC     JSB     COB     COB$CVTQP_R9
1448  2608
1448  2609 ; MODULE COB$CVTRPQ_R9 -- Rounded Packed to Quad conversion
1448  2610
1448  2611         MAC     JSB     COB     COB$CVTRPQ_R9
1450  2612
1450  2613 ; MODULE COB$CVTRQP_R9 -- Rounded Quad to Packed conversion
1450  2614         MAC     JSB     COB     COB$CVTRQP_R9
1458  2615
1458  2616
```

```
1458  2618 ; MODULE COB$MSGDEF -- Defines COB$ all conditon codes
1458  2619          MAC     SYM     COB     COB$_CALFAI
1458  2620          MAC     SYM     COB     COB$_CANFAIL
1458  2621          MAC     SYM     COB     COB$_DELINCOPE
1458  2622          MAC     SYM     COB     COB$_DELNO_R_S
1458  2623          MAC     SYM     COB     COB$_DELUNOFIL
1458  2624          MAC     SYM     COB     COB$_DISMORMAX
1458  2625          MAC     SYM     COB     COB$_EOFON_ACC
1458  2626          MAC     SYM     COB     COB$_ERRDURACC
1458  2627          MAC     SYM     COB     COB$_ERRDURDIS
1458  2628          MAC     SYM     COB     COB$_ERRDURSOR
1458  2629          MAC     SYM     COB     COB$_ERRON_FIL
1458  2630          MAC     SYM     COB     COB$_EXPDBOVER
1458  2631          MAC     SYM     COB     COB$_FAIFREEVM
1458  2632          MAC     SYM     COB     COB$_FAIGET_EF
1458  2633          MAC     SYM     COB     COB$_FAIGET_VM
1458  2634          MAC     SYM     COB     COB$_FATINTERR
1458  2635          MAC     SYM     COB     COB$_FILALRCLO
1458  2636          MAC     SYM     COB     COB$_FILALRLOC
1458  2637          MAC     SYM     COB     COB$_FILALROPE
1458  2638          MAC     SYM     COB     COB$_FILCLOLOC
1458  2639          MAC     SYM     COB     COB$_FILNOTFOU
1458  2640          MAC     SYM     COB     COB$_GOTO_ALT
1458  2641          MAC     SYM     COB     COB$_INTDIVZER
1458  2642          MAC     SYM     COB     COB$_INTEXPOVE
1458  2643          MAC     SYM     COB     COB$_INTEXPUND
1458  2644          MAC     SYM     COB     COB$_INTRESOPE
1458  2645          MAC     SYM     COB     COB$_INVARG
1458  2646          MAC     SYM     COB     COB$_INVCHANAM
1458  2647          MAC     SYM     COB     COB$_INVDECDIG
1458  2648          MAC     SYM     COB     COB$_INVLINVAL
1458  2649          MAC     SYM     COB     COB$_KEYNOTMAT
1458  2650          MAC     SYM     COB     COB$_LSTHNDLDB
1458  2651          MAC     SYM     COB     COB$_LSTHNDUSE
1458  2652          MAC     SYM     COB     COB$_NAMNOTLIN
1458  2653          MAC     SYM     COB     COB$_NESERRPER
1458  2654          MAC     SYM     COB     COB$_NORMAL
1458  2655          MAC     SYM     COB     COB$_NO_NEXLOG
1458  2656          MAC     SYM     COB     COB$_NO_NEXVAL
1458  2657          MAC     SYM     COB     COB$_NO_SPACE
1458  2658          MAC     SYM     COB     COB$_NO_USEPRO
1458  2659          MAC     SYM     COB     COB$_OCCDEPOVE
1458  2660          MAC     SYM     COB     COB$_OPTMISCLO
1458  2661          MAC     SYM     COB     COB$_OPTMISOPE
1458  2662          MAC     SYM     COB     COB$_OPTMISREA
1458  2663          MAC     SYM     COB     COB$_OPTMISSTA
1458  2664          MAC     SYM     COB     COB$_ORGNOTMAT
1458  2665          MAC     SYM     COB     COB$_PRIKEYCHA
1458  2666          MAC     SYM     COB     COB$_REAINCOPE
1458  2667          MAC     SYM     COB     COB$_REASMAMIN
1458  2668          MAC     SYM     COB     COB$_REAUNOFIL
1458  2669          MAC     SYM     COB     COB$_RECACTPER
1458  2670          MAC     SYM     COB     COB$_RECACTUSE
1458  2671          MAC     SYM     COB     COB$_RECLOCDEL
1458  2672          MAC     SYM     COB     COB$_RECLOCREA
1458  2673          MAC     SYM     COB     COB$_RECLOCREW
1458  2674          MAC     SYM     COB     COB$_RECLOCSTA
```

```
1458  2675          MAC     SYM     COB     COB$_RECLOCWRI
1458  2676          MAC     SYM     COB     COB$_RECLOC_OK
1458  2677          MAC     SYM     COB     COB$_RECNOTEXI
1458  2678          MAC     SYM     COB     COB$_RECNOTLOC
1458  2679          MAC     SYM     COB     COB$_REWCREDUP
1458  2680          MAC     SYM     COB     COB$_REWDISDUP
1458  2681          MAC     SYM     COB     COB$_REWINCOPE
1458  2682          MAC     SYM     COB     COB$_REWNO_R_S
1458  2683          MAC     SYM     COB     COB$_REWSMAMIN
1458  2684          MAC     SYM     COB     COB$_REWUNOFIL
1458  2685          MAC     SYM     COB     COB$_SETEXTFAI
1458  2686          MAC     SYM     COB     COB$_STAINCOPE
1458  2687          MAC     SYM     COB     COB$_STAUNOFIL
1458  2688          MAC     SYM     COB     COB$_SUBOVELON
1458  2689          MAC     SYM     COB     COB$_TIMOVELON
1458  2690          MAC     SYM     COB     COB$_UNDEF_EXP
1458  2691          MAC     SYM     COB     COB$_UNEINSCON
1458  2692          MAC     SYM     COB     COB$_UNLNO_CUR
1458  2693          MAC     SYM     COB     COB$_UNLUNOFIL
1458  2694          MAC     SYM     COB     COB$_WRIBEYBOU
1458  2695          MAC     SYM     COB     COB$_WRICREDUP
1458  2696          MAC     SYM     COB     COB$_WRIDISDUP
1458  2697          MAC     SYM     COB     COB$_WRIDUPALT
1458  2698          MAC     SYM     COB     COB$_WRIDUPKEY
1458  2699          MAC     SYM     COB     COB$_WRIINCOPE
1458  2700          MAC     SYM     COB     COB$_WRINOTASC
1458  2701          MAC     SYM     COB     COB$_WRISMAMIN
1458  2702          MAC     SYM     COB     COB$_WRIUNOFIL
```

```
1458  2704 ;+
1458  2705 ; The following routines are N O T in the sharable library.  They
1458  2706 ; are tabulated here to provide a complete tabulation of all entry
1458  2707 ; points known to COBOL when this module is assembled in the "ALLGBL"
1458  2708 ; mode.
1458  2709 ;-
1458  2710
1458  2711 ; MODULE COB$CALL -- Support COBOL CALL
1458  2712         MAC     NOVECT  COB     COB$CALL
1458  2713
1458  2714 ; MODULE COB$CANCEL -- Support COBOL CANCEL
1458  2715         MAC     NOVECT  COB     COB$CANCEL
1458  2716
1458  2717 ; MODULE COB$CNVOUT -- Support conversion routines
1458  2718         MAC     NOVECT  COB     COB$CNVOUT
1458  2719
1458  2720 ; MODULE COB$CVTDP -- Convert Double to Packed
1458  2721         MAC     NOVECT  COB     COB$CVTDP_R9
1458  2722
1458  2723 ; MODULE COB$CVTRDP -- Convert Rounded Double to Packed
1458  2724         MAC     NOVECT  COB     COB$CVTRDP_R9
1458  2725
1458  2726 ; MODULE COB$CVTPD -- Convert Packed to Double
1458  2727         MAC     NOVECT  COB     COB$CVTPD_R9
1458  2728
1458  2729 ; MODULE COB$CVTDQ -- Convert Double to Quadword
1458  2730         MAC     NOVECT  COB     COB$CVTDQ_R8
1458  2731
1458  2732 ; MODULE COB$CVTRDQ -- Convert Rounded Double to Quadword
1458  2733         MAC     NOVECT  COB     COB$CVTRDQ_R8
1458  2734
1458  2735 ; MODULE COB$CVTQD -- Convert Quadword to Double
1458  2736         MAC     NOVECT  COB     COB$CVTQD_R8
1458  2737
1458  2738 ; MODULE COB$CVTFP -- Convert Floating to Packed
1458  2739         MAC     NOVECT  COB     COB$CVTFP_R9
1458  2740
1458  2741 ; MODULE COB$CVTRFP -- Convert Rounded Floating to Packed
1458  2742         MAC     NOVECT  COB     COB$CVTRFP_R9
1458  2743
1458  2744 ; MODULE COB$CVTPF -- Convert Packed to Floaing
1458  2745         MAC     NOVECT  COB     COB$CVTPF_R9
1458  2746
1458  2747 ; MODULE COB$CVTFQ -- Convert Floating to Quadword
1458  2748         MAC     NOVECT  COB     COB$CVTFQ_R8
1458  2749
1458  2750 ; MODULE COB$CVTRFQ -- Convert Rounded Floating to Quadword
1458  2751         MAC     NOVECT  COB     COB$CVTRFQ_R8
1458  2752
1458  2753 ; MODULE COB$CVTQF -- Convert Quadword to Floating
1458  2754         MAC     NOVECT  COB     COB$CVTQF_R8
1458  2755
1458  2756 ; MODULE COB$EXPI -- CIT exponentiation
1458  2757         MAC     NOVECT  COB     COB$EXPI
1458  2758         MAC     NOVECT  COB     COB$EXPI_OSE
1458  2759
1458  2760 ; MODULE COB$LINAGE -- Support LINAGE
```

```
1458  2761          MAC     NOVECT  COB     COB$LINAGE
1458  2762          MAC     NOVECT  COB     COB$INIT_LINAGE
1458  2763          MAC     NOVECT  COB     COB$TERM_LINAGE
1458  2764
1458  2765 ; MODULE COB$RMS_BLOCKS -- RMS data block
1458  2766          MAC     NOVECT  COB     COB$AB_NAM
1458  2767
1458  2768 ; MODULE COB$SET_SWITCH -- Set external switches
1458  2769          MAC     NOVECT  COB     COB$SET_SWITCH
1458  2770
1458  2771 ; MODULE COB$SWITCH -- Support switches
1458  2772          MAC     NOVECT  COB     COB$SWITCH
1458  2773
1458  2774 ; MODULE LIB$AB_ASC_EBC -- ASCII to EBCDIC translation table
1458  2775          MAC     NOVECT  LIB     LIB$AB_ASC_EBC
1458  2776
1458  2777 ; MODULE LIB$AB_CVT_O_U -- Overpunch to Unsigned translation table
1458  2778          MAC     NOVECT  LIB     LIB$AB_CVT_O_U
1458  2779
1458  2780 ; MODULE LIB$AB_EBC_ASC -- EBCDIC to ASCII translation table
1458  2781          MAC     NOVECT  LIB     LIB$AB_EBC_ASC
1458  2782
```

```
1458  2784 ;       P O S T  V M S  V E R S I O N  2 . 0  A D D I T I O N S
1458  2785 ;       ---------------------------------------------------
1458  2786
1458  2787 ; This point marks where the modules added after VMS Version 2.0 are
1458  2788 ; placed (unless they fit in some existing hole).
1458  2789 ; The vector entries to the shared components come first, and
1458  2790 ; are then followed by the non-vectored (non-shared) entries and the new
1458  2791 ; symbol definitions.
1458  2792 ;       There are two flavors of shared components -- those that are
1458  2793 ; vectored and those that aren't.  Those that aren't are in the shared
1458  2794 ; image solely for the purpose of binding of VMSRTL.EXE, but are not
1458  2795 ; accessible from outside of the image.
1458  2796 ;
1458  2797 ; S H A R E D  C O M P O N E N T S  ( V E C T O R E D )
1458  2798 ; -------------------------------------------------
1458  2799
1458  2800
1458  2801 ; MODULE LIB$LUN
1458  2802         MAC     CALL    LIB     LIB$FREE_LUN
1460  2803         MAC     CALL    LIB     LIB$GET_LUN
1468  2804
1468  2805 ; MODULE LIB$EF
1468  2806         MAC     CALL    LIB     LIB$FREE_EF
147C  2807         MAC     CALL    LIB     LIB$GET_EF
1478  2808         MAC     CALL    LIB     LIB$RESERVE_EF
1480  2809
1480  2810 ; MODULE LIB$ANALYZE_SDESC
1480  2811         MAC     CALL    LIB     LIB$ANALYZE_SDESC
1488  2812         MAC     JSB     LIB     LIB$ANALYZE_SDESC_R2
1490  2813
1490  2814 ; MODULE STR$ANALYZE_SDESC
1490  2815         MAC     CALL    STR     STR$ANALYZE_SDESC
1498  2816         MAC     JSB     STR     STR$ANALYZE_SDESC_R1
14A0  2817
14A0  2818 ; MODULE LIB$FILESCAN -- find files matching wild-card description
14A0  2819         MAC     CALL    LIB     LIB$FILE_SCAN
14A8  2820         MAC     CALL    LIB     LIB$FIND_FILE
14B0  2821
14B0  2822
14B0  2823
14B0  2824 ; S H A R E D  C O M P O N E N T S  ( N O N - V E C T O R E D )
14B0  2825 ; -----------------------------------------------------------
14B0  2826
14B0  2827 ; MODULE STR$$CHECK_STATUS
14B0  2828         MAC     NOVECT  STR     STR$$CHECK_STATUS_R2
14B0  2829
14B0  2830
14B0  2831 ; N O N - S H A R E D  C O M P O N E N T S
14B0  2832 ; -----------------------------------------
14B0  2833
14B0  2834 ; MODULE LIB$AB_ASC_EBC_REV -- Reversible ASCII to EBCDIC trans. table
14B0  2835         MAC     NOVECT  LIB     LIB$AB_ASC_EBC_REV
14B0  2836
14B0  2837 ; MODULE LIB$AB_EBC_ASC_REV -- Reversible EBCDIC to ASCII trans. table
14B0  2838         MAC     NOVECT  LIB     LIB$AB_EBC_ASC_REV
14B0  2839
14B0  2840 ; MODULE LIB$AB_CVTPT_Z -- packed decimal to zoned translation table
```

```
14B0  2841          MAC     NOVECT  LIB     LIB$AB_CVTPT_Z
14B0  2842
14B0  2843 ; MODULE LIB$AB_CVTTP_Z -- zoned to packed decimal translation table
14B0  2844          MAC     NOVECT  LIB     LIB$AB_CVTTP_Z
14B0  2845
14B0  2846 ; MODULE LIB$CALLG -- execute CALLG instruction
14B0  2847          MAC     NOVECT  LIB     LIB$CALLG
14B0  2848
14B0  2849 ; MODULE LIB$DECODE_FAULT -- decode instruction stream
14B0  2850          MAC     NOVECT  LIB     LIB$DECODE_FAULT
14B0  2851
14B0  2852 ; MODULE LIB$EDIV -- execute EDIV instruction
14B0  2853          MAC     NOVECT  LIB     LIB$EDIV
14B0  2854
14B0  2855 ; MODULE LIB$EMUL -- execute EMUL instruction
14B0  2856          MAC     NOVECT  LIB     LIB$EMUL
14B0  2857
14B0  2858 ; MODULE LIB$MOVC3 -- execute MOVC3 instruction
14B0  2859          MAC     NOVECT  LIB     LIB$MOVC3
14B0  2860
14B0  2861 ; MODULE LIB$MOVC5 -- execute MOVC5 instruction
14B0  2862          MAC     NOVECT  LIB     LIB$MOVC5
14B0  2863
14B0  2864 ; MODULE COB$AB_DEEDIT -- COBOL translation table for "de-editing
14B0  2865          MAC     NOVECT  COB     COB$AB_DEEDIT
14B0  2866
14B0  2867 ; MODULE COB$DBEXCEPTION -- COBOL Data Base Exception Processing
14B0  2868          MAC     NOVECT  COB     COB$DBEXCEPTION
14B0  2869
14B0  2870 ; MODULE COB$AB_SPANC2 -- COBOL  SPANC translation table II
14B0  2871          MAC     NOVECT  COB     COB$AB_SPANC2
14B0  2872
14B0  2873 ; MODULE LIB$CVTDXDX -- LIB general data type conversion routine
14B0  2874          MAC     NOVECT  LIB     LIB$CVT_DX_DX
14B0  2875
14B0  2876 ; MODULE LIB$$PACK_ARITH -- perform packed arithmetic for STR$DIVIDE
14B0  2877          MAC     NOVECT  LIB     LIB$$CVT_STR_PACK_R9
14B0  2878          MAC     NOVECT  LIB     LIB$$CALC_D_R7
14B0  2879          MAC     NOVECT  LIB     LIB$$CALC_Q_R9
14B0  2880          MAC     NOVECT  LIB     LIB$$ADJUST_Q_R9
14B0  2881          MAC     NOVECT  LIB     LIB$$MUL_PACK_R10
14B0  2882          MAC     NOVECT  LIB     LIB$$SUB_PACK_R8
14B0  2883          MAC     NOVECT  LIB     LIB$$ROUND_R7
14B0  2884          MAC     NOVECT  LIB     LIB$$CVT_PACK_STR_R8
14B0  2885
14B0  2886 ; MODULE STR$ARITH -- string arithmetic (added entry point STR$DIVIDE)
14B0  2887          MAC     NOVECT  STR     STR$DIVIDE
14B0  2888
14B0  2889 ; MODULE FOR$INIUND -- FORTRAN underflow handler initialization
14B0  2890          MAC     NOVECT  FOR     FOR$INIT_UNDER
14B0  2891
14B0  2892 ; MODULE FOR$UNDERF -- FORTRAN underflow handler
14B0  2893          MAC     NOVECT  FOR     FOR$UNDERFLOW_HANDLER
14B0  2894
14B0  2895 ; MODULE OTS$POWLULU -- unsigned ** unsigned integer power
14B0  2896          MAC     NOVECT  OTS     OTS$POWLOLU
14B0  2897
```

```
                14B0  2898 ; MODULE STR$COMPARE_CASE_BLIND -- Compare strings case-blind
                14B0  2899         MAC     NOVECT  STR     STR$CASE_BLIND_COMPARE
                14B0  2900
                14B0  2901 ; MODULE STR$FIND_FIRST -- Find 1st char in or not in set
                14B0  2902         MAC     NOVECT  STR     STR$FIND_FIRST_IN_SET
                14B0  2903         MAC     NOVECT  STR     STR$FIND_FIRST_NOT_IN_SET
                14B0  2904
                14B0  2905 ; MODULE STR$FIND_FIRST_SUBSTRING -- Find first substring
                14B0  2906         MAC     NOVECT  STR     STR$FIND_FIRST_SUBSTRING
                14B0  2907
                14B0  2908 ; N E W  E N T R Y  P O I N T S  F O R  V A X  B A S I C  2.0
                14B0  2909 ; ------------------------------------------------------------
                14B0  2910
                14B0  2911 ; MODULE BAS$CVTTP -- Basic convert text to packed
                14B0  2912         MAC     CALL    BAS     BAS$CVT_T_P
                14B8  2913
                14B8  2914 ; MODULE LIB$$ADDP -- add packed instruction for BAS$CVT_T_P
                14B8  2915         MAC     NOVECT  LIB     LIB$$ADDP_R7
                14B8  2916
                14B8  2917 ; MODULE BAS$UPI_TERM_IO
                14B8  2918 ;   (use up addr vacated by LIB$$ADDP_R7)
                14B8  2919         MAC     CALL    BAS     BAS$IN_B_R
                14C0  2920
                14C0  2921 ; MODULE OTS$CNVOUT -- convert floating to E formatted text
                14C0  2922 ;   (shared, not vectored)
                14C0  2923         MAC     NOVECT  OTS     OTS$CNVOUT
                14C0  2924
                14C0  2925 ; OLD ENTRY POINTS FOR MODULE BAS$CVTOUT, originally overlooked
                14C0  2926         MAC     CALL    BAS     BAS$CVT_OUT_F_E
                14C8  2927         MAC     CALL    BAS     BAS$CVT_OUT_F_F
                14D0  2928         MAC     CALL    BAS     BAS$CVT_OUT_D_E
                14D8  2929         MAC     CALL    BAS     BAS$CVT_OUT_D_F
                14E0  2930         MAC     CALL    BAS     BAS$CVT_OUT_D_G
                14E8  2931
                14E8  2932 ; NEW ENTRY POINTS FOR MODULE BAS$CVTOUT, Basic output conversion
                14E8  2933         MAC     CALL    BAS     BAS$CVT_OUT_G_E
                14F0  2934         MAC     CALL    BAS     BAS$CVT_OUT_G_F
                14F8  2935         MAC     CALL    BAS     BAS$CVT_OUT_G_G
                1500  2936         MAC     CALL    BAS     BAS$CVT_OUT_H_E
                1508  2937         MAC     CALL    BAS     BAS$CVT_OUT_H_F
                1510  2938         MAC     CALL    BAS     BAS$CVT_OUT_H_G
                1518  2939         MAC     CALL    BAS     BAS$CVT_OUT_P_E
                1520  2940         MAC     CALL    BAS     BAS$CVT_OUT_P_F
                1528  2941         MAC     CALL    BAS     BAS$CVT_OUT_P_G
                1530  2942
                1530  2943 ; NEW ENTRY POINTS FOR MODULE BAS$CMPAPP, Basic compare approximate
                1530  2944         MAC     CALL    BAS     BAS$CMPG_APP
                1538  2945         MAC     CALL    BAS     BAS$CMPH_APP
                1540  2946
                1540  2947 ; NEW ENTRY POINTS FOR MODULE BAS$$COPYFD, Basic copy floating
                1540  2948 ;   (shared, not vectored)
                1540  2949         MAC     NOVECT  BAS     BAS$$COPY_G_R1
                1540  2950         MAC     NOVECT  BAS     BAS$$COPY_H_R3
                1540  2951
                1540  2952 ; NEW ENTRY POINTS FOR MODULE BAS$NUM, Basic NUM function
                1540  2953         MAC     CALL    BAS     BAS$NUM_G
                1548  2954         MAC     CALL    BAS     BAS$NUM_H
```

```
1550  2955          MAC     CALL    BAS     BAS$NUM_P
1558  2956
1558  2957 ; NEW ENTRY POINTS FOR MODULE BAS$NUM1, Basic NUM1 function
1558  2958          MAC     CALL    BAS     BAS$NUM1_G
1560  2959          MAC     CALL    BAS     BAS$NUM1_H
1568  2960          MAC     CALL    BAS     BAS$NUM1_P
1570  2961
1570  2962 ; NEW ENTRY POINTS FOR MODULE BAS$STR, Basic STR$ function
1570  2963          MAC     CALL    BAS     BAS$STR_G
1578  2964          MAC     CALL    BAS     BAS$STR_H
1580  2965          MAC     CALL    BAS     BAS$STR_P
1588  2966
1588  2967 ; NEW ENTRY POINTS FOR MODULE BAS$UPI_TERM_IO, Basic UPI level I/O
1588  2968          MAC     CALL    BAS     BAS$OUT_G_V_S
1590  2969          MAC     CALL    BAS     BAS$OUT_G_V_B
1598  2970          MAC     CALL    BAS     BAS$OUT_G_V_C
15A0  2971          MAC     CALL    BAS     BAS$OUT_H_V_S
15A8  2972          MAC     CALL    BAS     BAS$OUT_H_V_B
15B0  2973          MAC     CALL    BAS     BAS$OUT_H_V_C
15B8  2974          MAC     CALL    BAS     BAS$OUT_P_DX_S
15C0  2975          MAC     CALL    BAS     BAS$OUT_P_DX_B
15C8  2976          MAC     CALL    BAS     BAS$OUT_P_DX_C
15D0  2977          MAC     CALL    BAS     BAS$IN_G_R
15D8  2978          MAC     CALL    BAS     BAS$IN_H_R
15E0  2979          MAC     CALL    BAS     BAS$IN_P_DX
15E8  2980
15E8  2981 ; NEW ENTRY POINTS FOR BAS$VAL, Basic VAL function
15E8  2982          MAC     CALL    BAS     BAS$VAL_G
15F0  2983          MAC     CALL    BAS     BAS$VAL_H
15F8  2984          MAC     CALL    BAS     BAS$VAL_P
1600  2985
1600  2986 ; MODULE BAS$$REC_PROC
1600  2987 ;   this is needed for BAS$ANSI_TAB, a non-shared entry point
1600  2988          MAC     JSB     BAS     BAS$$REC_WSL1
1608  2989
1608  2990 ; MODULE BASFIND, new entry point
1608  2991          MAC     CALL    BAS     BAS$FIND_RFA
1610  2992
1610  2993 ; MODULE BASGET, new entry point
1610  2994          MAC     CALL    BAS     BAS$GET_RFA
1618  2995
1618  2996 ; MODULE BASGETRFA, new
1618  2997          MAC     CALL    BAS     BAS$GETRFA
1620  2998
1620  2999 ; MODULE BASCB, old entry point must be vectored for improved BASKILL
1620  3000          MAC     CALL    BAS     BAS$$NEXT_LUN
1628  3001
1628  3002 ; MODULE BASIOBEG, new entry point
1628  3003          MAC     CALL    BAS     BAS$ANSI_INPUT
1630  3004
1630  3005 ; MODULE BASIOEND, new entry point
1630  3006          MAC     CALL    BAS     BAS$ANSI_IO_END
1638  3007
1638  3008 ; MODULE BASCTRLC, all entry points
1638  3009          MAC     CALL    BAS     BAS$CTRLC
1640  3010          MAC     CALL    BAS     BAS$RCTRLC
1648  3011          MAC     CALL    BAS     BAS$$CTRLC_INIT
```

1650 3012

```
                        1650  3014          .SBTTL  MTH$$AB_ALOG - Table for ALOG routines
                        1650  3015
                        1650  3016 ;
                        1650  3017 ;   The MTH$$AB_ALOG table is accessed by the low order exponent  bit  and the
                        1650  3018 ;   first 7 fraction bits (not including the hidden bit) of the argument.  The
                        1650  3019 ;   table  entries  are  offsets  into  the  F_FHI table.  Note that the
                        1650  3020 ;   MTH$$AB_ALOG table is data type independent and is used by  all  four  LOG
                        1650  3021 ;   routines.
                        1650  3022 ;
                        1650  3023 ;   This table is a duplicate of that in MTHALOG.MAR, but must remain
                        1650  3024 ;   separate.
                        1650  3025
                        1650  3026
                        1650  3027 MTH$$AB_ALOG:
04 04 04 04 04 04 04 04 1650  3028          .BYTE   ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04
04 04 04 04 04 04 04 04 1658  3029          .BYTE   ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04
09 09 09 04 04 04 04 04 1660  3030          .BYTE   ^X04, ^X04, ^X04, ^X04, ^X04, ^X09, ^X09, ^X09
09 09 09 09 09 09 09 09 1668  3031          .BYTE   ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09
09 09 09 09 09 09 09 09 1670  3032          .BYTE   ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09
0E 0E 0E 0E 0E 0E 0E 0E 1678  3033          .BYTE   ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E
0E 0E 0E 0E 0E 0E 0E 0E 1680  3034          .BYTE   ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E
13 13 13 13 13 13 13 13 1688  3035          .BYTE   ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X13
18 18 13 13 13 13 13 13 1690  3036          .BYTE   ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X18, ^X18
18 18 18 18 18 18 18 18 1698  3037          .BYTE   ^X18, ^X18, ^X18, ^X18, ^X18, ^X18, ^X18, ^X18
1D 1D 1D 1D 1D 1D 1D 18 16A0  3038          .BYTE   ^X18, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D
22 22 22 22 22 22 1D 1D 16A8  3039          .BYTE   ^X1D, ^X1D, ^X22, ^X22, ^X22, ^X22, ^X22, ^X22
2C 27 27 27 27 27 27 22 16B0  3040          .BYTE   ^X22, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27, ^X2C
31 31 31 31 2C 2C 2C 2C 16B8  3041          .BYTE   ^X2C, ^X2C, ^X2C, ^X2C, ^X31, ^X31, ^X31, ^X31
45 40 40 3B 3B 36 36 36 16C0  3042          .BYTE   ^X36, ^X36, ^X36, ^X3B, ^X3B, ^X40, ^X40, ^X45
FF FF FF FF FF FF FF FF 16C8  3043          .BYTE   ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF
32 37 3C 41 FF FF FF FF 16D0  3044          .BYTE   ^XFF, ^XFF, ^XFF, ^XFF, ^X41, ^X3C, ^X37, ^X32
23 23 28 28 28 2D 2D 32 16D8  3045          .BYTE   ^X32, ^X2D, ^X2D, ^X28, ^X28, ^X28, ^X23, ^X23
19 1E 1E 1E 1E 1E 23 23 16E0  3046          .BYTE   ^X23, ^X23, ^X1E, ^X1E, ^X1E, ^X1E, ^X1E, ^X19
14 14 19 19 19 19 19 19 16E8  3047          .BYTE   ^X19, ^X19, ^X19, ^X19, ^X19, ^X19, ^X14, ^X14
0F 14 14 14 14 14 14 14 16F0  3048          .BYTE   ^X14, ^X14, ^X14, ^X14, ^X14, ^X14, ^X14, ^X0F
0F 0F 0F 0F 0F 0F 0F 0F 16F8  3049          .BYTE   ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F
0A 0A 0A 0A 0A 0F 0F 0F 1700  3050          .BYTE   ^X0F, ^X0F, ^X0F, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A
0A 0A 0A 0A 0A 0A 0A 0A 1708  3051          .BYTE   ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A
05 05 05 05 0A 0A 0A 0A 1710  3052          .BYTE   ^X0A, ^X0A, ^X0A, ^X0A, ^X05, ^X05, ^X05, ^X05
05 05 05 05 05 05 05 05 1718  3053          .BYTE   ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05
05 05 05 05 05 05 05 05 1720  3054          .BYTE   ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05
00 00 00 00 05 05 05 05 1728  3055          .BYTE   ^X05, ^X05, ^X05, ^X05, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1730  3056          .BYTE   ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1738  3057          .BYTE   ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1740  3058          .BYTE   ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1748  3059          .BYTE   ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
```

```
                                1750  3061          .SBTTL  MTH$$AB_ATAN - Table for ATAN routines
                                1750  3062
                                1750  3063 ;
                                1750  3064 ;   The MTH$$AB_ATAN table is a table of byte entries used to obtain an index
                                1750  3065 ;   into the ATAN_TABLE.  MTH$$AB_ATAN is indexed using the low order bits of
                                1750  3066 ;   the  exponent  field  and the high order bits of the fraction field.  The
                                1750  3067 ;   MTH$$AB_ATAN table is independent of the data type and is used by all  of
                                1750  3068 ;   the arctangent routines.
                                1750  3069 ;
                                1750  3070 ;   This table is a duplicate of that in MTHATAN.MAR, but must remain
                                1750  3071 ;   separate.
                                1750  3072
                                1750  3073 MTH$$AB_ATAN:
09 06 06 03 03 00 00 00         1750  3074          .BYTE   ^X00, ^X00, ^X00, ^X03, ^X03, ^X06, ^X06, ^X09
12 0F 0F 0C 0C 0C 09 09         1758  3075          .BYTE   ^X09, ^X09, ^X0C, ^X0C, ^X0C, ^X0F, ^X0F, ^X12
18 18 18 15 15 15 12 12         1760  3076          .BYTE   ^X12, ^X12, ^X15, ^X15, ^X15, ^X18, ^X18, ^X18
21 1E 1E 1E 1B 1B 1B 1B         1768  3077          .BYTE   ^X1B, ^X1B, ^X1B, ^X1B, ^X1E, ^X1E, ^X1E, ^X21
24 24 24 24 21 21 21 21         1770  3078          .BYTE   ^X21, ^X21, ^X21, ^X21, ^X24, ^X24, ^X24, ^X24
27 27 27 27 27 27 24 24         1778  3079          .BYTE   ^X24, ^X24, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27
   27 27 27 27 27 27 27         1780  3080          .BYTE   ^X27, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27
                                1787  3081
                                1787  3082          .END
```

```
$$BAS$$BLNK_LINE                00000BA0 RG  01
$$BAS$$CB_GET                   00000B80 RG  01
$$BAS$$CB_POP                   00000B70 RG  01
$$BAS$$CB_PUSH                  00000B78 RG  01
$$BAS$$CLOSE_ALL                00000BF8 RG  01
$$BAS$$CTRLC_INIT               00001648 RG  01
$$BAS$$ERR_INIT                 00000B88 RG  01
$$BAS$$FORMAT_INT               00000BF0 RG  01
$$BAS$$NEXT_LUN                 00001620 RG  01
$$BAS$$OPEN_ZERO                00000B90 RG  01
$$BAS$$RECOO_INIT               00000B98 RG  01
$$BAS$$REC_WSL1                 00001600 RG  01
$$BAS$$SCALE_L_R1               00000BD8 RG  01
$$BAS$$SCALE_RT                 00000BE0 RG  01
$$BAS$$SIGNAL                   00000BA8 RG  01
$$BAS$$SIGNAL_IO                00000BB0 RG  01
$$BAS$$STATU_INIT               00000BB8 RG  01
$$BAS$$STOP                     00000BC0 RG  01
$$BAS$$STOP_IO                  00000BC8 RG  01
$$BAS$$STOP_RMS                 00000BE8 RG  01
$$BAS$$UDF_RL1                  00000C00 RG  01
$$BAS$$UDF_WL1                  00000C08 RG  01
$$BAS$ANSI_INPUT                00001628 RG  01
$$BAS$ANSI_IO_END               00001630 RG  01
$$BAS$BUFSIZ                    00000B40 RG  01
$$BAS$CANTYPAHEAD               00000BD0 RG  01
$$BAS$CCPOS                     00000B20 RG  01
$$BAS$CHR                       00000B48 RG  01
$$BAS$CLOSE                     00000A88 RG  01
$$BAS$CMPD_APP                  00000890 RG  01
$$BAS$CMPF_APP                  00000888 RG  01
$$BAS$CMPG_APP                  00001530 RG  01
$$BAS$CMPH_APP                  00001538 RG  01
$$BAS$CTRLC                     00001638 RG  01
$$BAS$CVT_OUT_D_E               000014D0 RG  01
$$BAS$CVT_OUT_D_F               000014D8 RG  01
$$BAS$CVT_OUT_D_G               000014E0 RG  01
$$BAS$CVT_OUT_F_E               000014C0 RG  01
$$BAS$CVT_OUT_F_F               000014C8 RG  01
$$BAS$CVT_OUT_G_E               000014E8 RG  01
$$BAS$CVT_OUT_G_F               000014F0 RG  01
$$BAS$CVT_OUT_G_G               000014F8 RG  01
$$BAS$CVT_OUT_H_E               00001500 RG  01
$$BAS$CVT_OUT_H_F               00001508 RG  01
$$BAS$CVT_OUT_H_G               00001510 RG  01
$$BAS$CVT_OUT_P_E               00001518 RG  01
$$BAS$CVT_OUT_P_F               00001520 RG  01
$$BAS$CVT_OUT_P_G               00001528 RG  01
$$BAS$CVT_T_P                   000014B0 RG  01
$$BAS$DELETE                    00000AE0 RG  01
$$BAS$DSCALE_D_R1               00000880 RG  01
$$BAS$EDIT                      000008A8 RG  01
$$BAS$END_DEF_R8                00000940 RG  01
$$BAS$END_DFS_R8                00000948 RG  01
$$BAS$END_GSB_R8                00000950 RG  01
$$BAS$END_R8                    00000938 RG  01
$$BAS$ERL                       00000980 RG  01
$$BAS$ERN                       00000988 RG  01
$$BAS$ERR                       00000978 RG  01
$$BAS$ERROR                     00000B50 RG  01
$$BAS$ERT                       00000990 RG  01
$$BAS$FIND                      00000AC8 RG  01
$$BAS$FIND_KEY                  00000AD8 RG  01
$$BAS$FIND_RECORD               00000AD0 RG  01
$$BAS$FIND_RFA                  00001608 RG  01
$$BAS$FREE                      00000B18 RG  01
$$BAS$GET                       00000A90 RG  01
$$BAS$GETRFA                    00001618 RG  01
$$BAS$GET_KEY                   00000AA0 RG  01
$$BAS$GET_RECORD                00000A98 RG  01
$$BAS$GET_RFA                   00001610 RG  01
$$BAS$HANDLER                   00000998 RG  01
$$BAS$INIT_DEF_R8               00000920 RG  01
$$BAS$INIT_DFS_R8               00000928 RG  01
$$BAS$INIT_GOSOB                00000930 RG  01
$$BAS$INIT_R8                   00000918 RG  01
$$BAS$INPUT                     000009A0 RG  01
$$BAS$INPUT_LINE                000009B0 RG  01
$$BAS$INSTR                     000008B0 RG  01
$$BAS$IN_B_R                    000014B8 RG  01
$$BAS$IN_D_R                    000009F0 RG  01
$$BAS$IN_F_R                    000009E8 RG  01
$$BAS$IN_G_R                    000015D0 RG  01
$$BAS$IN_H_R                    000015D8 RG  01
$$BAS$IN_L_R                    000009E0 RG  01
$$BAS$IN_P_DX                   000015E0 RG  01
$$BAS$IN_T_DX                   000009F8 RG  01
$$BAS$IN_W_R                    000009D8 RG  01
$$BAS$IO_END                    000009D0 RG  01
$$BAS$LINPUT                    000009A8 RG  01
$$BAS$MAT_INPUT                 00000A68 RG  01
$$BAS$MAT_LINPUT                00000A70 RG  01
$$BAS$MAT_PRINT                 00000A60 RG  01
$$BAS$MAT_READ                  00000A78 RG  01
$$BAS$NUMT_D                    000008F0 RG  01
$$BAS$NUM1_F                    000008E8 RG  01
$$BAS$NUM1_G                    00001558 RG  01
$$BAS$NUM1_H                    00001560 RG  01
$$BAS$NUM1_L                    000008F8 RG  01
$$BAS$NUM1_P                    00001568 RG  01
$$BAS$NUM_D                     000008D8 RG  01
$$BAS$NUM_F                     000008D0 RG  01
$$BAS$NUM_G                     00001540 RG  01
$$BAS$NUM_H                     00001548 RG  01
$$BAS$NUM_L                     000008E0 RG  01
$$BAS$NUM_P                     00001550 RG  01
$$BAS$ON_ERR_BK                 00000960 RG  01
$$BAS$ON_ERR_Z                  00000958 RG  01
$$BAS$OPEN                      00000A80 RG  01
$$BAS$OUT_D_V_B                 00000A38 RG  01
$$BAS$OUT_D_V_C                 00000A40 RG  01
$$BAS$OUT_D_V_S                 00000A30 RG  01
$$BAS$OUT_F_V_B                 00000A20 RG  01
$$BAS$OUT_F_V_C                 00000A28 RG  01
```

```
$$BAS$OUT_F_V_S                 00000A18 RG  01        $$COB$ACC_TIME                  00001400 RG  01
$$BAS$OUT_G_V_B                 00001590 RG  01        $$COB$ADDI                      00001330 RG  01
$$BAS$OUT_G_V_C                 00001598 RG  01        $$COB$CMPI                      00001358 RG  01
$$BAS$OUT_G_V_S                 00001588 RG  01        $$COB$CVTDI_R7                  00001360 RG  01
$$BAS$OUT_H_V_B                 000015A8 RG  01        $$COB$CVTFI_R7                  00001368 RG  01
$$BAS$OUT_H_V_C                 000015B0 RG  01        $$COB$CVTID_R7                  00001370 RG  01
$$BAS$OUT_H_V_S                 000015A0 RG  01        $$COB$CVTIF_R7                  00001378 RG  01
$$BAS$OUT_L_V_B                 00000A08 RG  01        $$COB$CVTIL_R8                  00001380 RG  01
$$BAS$OUT_L_V_C                 00000A10 RG  01        $$COB$CVTIP_R9                  00001388 RG  01
$$BAS$OUT_L_V_S                 00000A00 RG  01        $$COB$CVTIQ_R8                  00001390 RG  01
$$BAS$OUT_P_DX_B                000015C0 RG  01        $$COB$CVTIW_R8                  00001398 RG  01
$$BAS$OUT_P_DX_C                000015C8 RG  01        $$COB$CVTLI_R8                  000013A0 RG  01
$$BAS$OUT_P_DX_S                000015B8 RG  01        $$COB$CVTPI_R9                  000013A8 RG  01
$$BAS$OUT_T_DX_B                00000A50 RG  01        $$COB$CVTPQ_R9                  00001438 RG  01
$$BAS$OUT_T_DX_C                00000A58 RG  01        $$COB$CVTQI_R8                  000013B0 RG  01
$$BAS$OUT_T_DX_S                00000A48 RG  01        $$COB$CVTQP_R9                  00001440 RG  01
$$BAS$POP_ERR                   00000B68 RG  01        $$COB$CVTRIC_R8                 000013B8 RG  01
$$BAS$PRINT                     000009C0 RG  01        $$COB$CVTRIP_R9                 000013C0 RG  01
$$BAS$PRINT_USING               000009C8 RG  01        $$COB$CVTRIQ_R8                 000013C8 RG  01
$$BAS$PUSH_ERR                  00000B60 RG  01        $$COB$CVTRIW_R8                 000013D0 RG  01
$$BAS$PUT                       00000AA8 RG  01        $$COB$CVTRPQ_R9                 00001448 RG  01
$$BAS$PUT_COUNT                 00000AB8 RG  01        $$COB$CVTRQP_R9                 00001450 RG  01
$$BAS$PUT_RECORD                00000AB0 RG  01        $$COB$CVTTI_R8                  000013D8 RG  01
$$BAS$PUT_REC_CNT               00000AC0 RG  01        $$COB$CVTWI_R8                  000013E0 RG  01
$$BAS$RCTRLC                    00001640 RG  01        $$COB$DISPLAY                   00001410 RG  01
$$BAS$READ                      000009B8 RG  01        $$COB$DISP_NO_ADV               00001418 RG  01
$$BAS$RECOUNT                   00000B30 RG  01        $$COB$DIVI                      00001348 RG  01
$$BAS$RESTORE                   00000AF8 RG  01        $$COB$DIVI_OSE                  00001350 RG  01
$$BAS$RESTORE_DAT               00000B28 RG  01        $$COB$DIVQ_R8                   00001420 RG  01
$$BAS$RESTORE_KEY               00000B00 RG  01        $$COB$ERROR                     00001328 RG  01
$$BAS$RESUME                    00000968 RG  01        $$COB$HANDLER                   00001318 RG  01
$$BAS$RESUME_Z                  00000970 RG  01        $$COB$IOEXCEPTION               00001320 RG  01
$$BAS$RSET                      00000898 RG  01        $$COB$MULI                      00001340 RG  01
$$BAS$RSET_R                    000008A0 RG  01        $$COB$MULQ_R8                   00001428 RG  01
$$BAS$SCALE_D_R1                00000878 RG  01        $$COB$PAUSE                     00001430 RG  01
$$BAS$SCRATCH                   00000B08 RG  01        $$COB$SUBI                      00001338 RG  01
$$BAS$STATUS                    00000B38 RG  01        $$FOR$$CB_GET                   00000620 RG  01
$$BAS$STR_D                     000008C0 RG  01        $$FOR$$CB_POP                   00000610 RG  01
$$BAS$STR_F                     00000888 RG  01        $$FOR$$CB_PUSH                  00000608 RG  01
$$BAS$STR_G                     00001570 RG  01        $$FOR$$CB_RET                   00000618 RG  01
$$BAS$STR_H                     00001578 RG  01        $$FOR$$ERRSNS_SAV               00000628 RG  01
$$BAS$STR_L                     000008C8 RG  01        $$FOR$$FP_MATCH                 00000728 RG  01
$$BAS$STR_P                     00001580 RG  01        $$FOR$BACKSPACE                 00000180 RG  01
$$BAS$UNLOCK                    00000B10 RG  01        $$FOR$CLOSE                     00000000 RG  01
$$BAS$UPDATE                    00000AE8 RG  01        $$FOR$CNV_IN_DEFG               00000200 RG  01
$$BAS$UPDATE_COUN               00000AF0 RG  01        $$FOR$CNV_IN_I                  00000210 RG  01
$$BAS$VAL_D                     00000910 RG  01        $$FOR$CNV_IN_L                  00000218 RG  01
$$BAS$VAL_F                     00000908 RG  01        $$FOR$CNV_IN_O                  00000220 RG  01
$$BAS$VAL_G                     000015E8 RG  01        $$FOR$CNV_IN_Z                  00000228 RG  01
$$BAS$VAL_H                     000015F0 RG  01        $$FOR$CNV_OUT_D                 000001A8 RG  01
$$BAS$VAL_L                     00000900 RG  01        $$FOR$CNV_OUT_E                 000001B0 RG  01
$$BAS$VAL_P                     000015F8 RG  01        $$FOR$CNV_OUT_F                 000001B8 RG  01
$$BAS$WAIT                      00000CA0 RG  01        $$FOR$CNV_OUT_G                 000001C0 RG  01
$$COB$ACCEPT                    00001408 RG  01        $$FOR$CNV_OUT_I                 00000188 RG  01
$$COB$ACC_DATE                  000013E8 RG  01        $$FOR$CNV_OUT_L                 00000190 RG  01
$$COB$ACC_DAY                   000013F0 RG  01        $$FOR$CNV_OUT_O                 00000198 RG  01
$$COB$ACC_DAYWEEK               000013F8 RG  01        $$FOR$CNV_OUT_Z                 000001A0 RG  01
```

```
$$FOR$CVT_D_TD                  000001A8 RG    01
$$FOR$CVT_D_TE                  000001B0 RG    01
$$FOR$CVT_D_TF                  000001B8 RG    01
$$FOR$CVT_D_TG                  000001C0 RG    01
$$FOR$CVT_G_TD                  00000640 RG    01
$$FOR$CVT_G_TE                  00000648 RG    01
$$FOR$CVT_G_TF                  00000650 RG    01
$$FOR$CVT_G_TG                  00000658 RG    01
$$FOR$CVT_H_TD                  00000668 RG    01
$$FOR$CVT_H_TE                  00000670 RG    01
$$FOR$CVT_H_TF                  00000678 RG    01
$$FOR$CVT_H_TG                  00000680 RG    01
$$FOR$DECODE_MF                 00000008 RG    01
$$FOR$DECODE_MO                 00000010 RG    01
$$FOR$DEF_FILE                  000001C8 RG    01
$$FOR$DEF_FILE_W                000001D0 RG    01
$$FOR$DELETE                    000006E8 RG    01
$$FOR$DELETE_D                  000006F0 RG    01
$$FOR$ENCODE_MF                 00000018 RG    01
$$FOR$ENCODE_MO                 00000020 RG    01
$$FOR$ENDFILE                   000001D8 RG    01
$$FOR$ERRSNS                    000001E0 RG    01
$$FOR$ERRSNS_W                  000001E8 RG    01
$$FOR$EXIT                      000001F0 RG    01
$$FOR$EXIT_W                    000001F8 RG    01
$$FOR$FIND                      00000208 RG    01
$$FOR$INI_DES1_R2               00000230 RG    01
$$FOR$INI_DES2_R3               00000238 RG    01
$$FOR$INI_DESC_R6               00000240 RG    01
$$FOR$INQUIRE                   000006F8 RG    01
$$FOR$IO_B_R                    000000E0 RG    01
$$FOR$IO_B_V                    000000E8 RG    01
$$FOR$IO_DC_R                   00000128 RG    01
$$FOR$IO_DC_V                   00000630 RG    01
$$FOR$IO_D_R                    000000C0 RG    01
$$FOR$IO_D_V                    000000C8 RG    01
$$FOR$IO_END                    000000A8 RG    01
$$FOR$IO_FC_R                   00000140 RG    01
$$FOR$IO_FC_V                   00000148 RG    01
$$FOR$IO_F_R                    000000B0 RG    01
$$FOR$IO_F_V                    000000B8 RG    01
$$FOR$IO_GC_R                   00000130 RG    01
$$FOR$IO_GC_V                   00000638 RG    01
$$FOR$IO_G_R                    00000108 RG    01
$$FOR$IO_G_V                    00000110 RG    01
$$FOR$IO_H_R                    00000118 RG    01
$$FOR$IO_H_V                    00000120 RG    01
$$FOR$IO_LO_R                   00000150 RG    01
$$FOR$IO_LU_V                   00000158 RG    01
$$FOR$IO_L_R                    000000D0 RG    01
$$FOR$IO_L_V                    000000D8 RG    01
$$FOR$IO_T_DS                   000000F0 RG    01
$$FOR$IO_T_V_DS                 00000138 RG    01
$$FOR$IO_WO_R                   00000160 RG    01
$$FOR$IO_WU_V                   00000168 RG    01
$$FOR$IO_W_R                    000000F8 RG    01
$$FOR$IO_W_V                    00000100 RG    01
$$FOR$IO_X_DA                   00000170 RG    01
$$FOR$IO_X_NL                   00000748 RG    01
$$FOR$IO_X_SB                   00000740 RG    01
$$FOR$IO_X_SE                   00000750 RG    01
$$FOR$OPEN                      00000178 RG    01
$$FOR$PAUSE                     00000248 RG    01
$$FOR$RAB                       00000788 RG    01
$$FOR$READ_DF                   00000038 RG    01
$$FOR$READ_DO                   00000040 RG    01
$$FOR$READ_DU                   00000048 RG    01
$$FOR$READ_IF                   000006C8 RG    01
$$FOR$READ_IO                   000006D0 RG    01
$$FOR$READ_KF                   00000028 RG    01
$$FOR$READ_KO                   00000030 RG    01
$$FOR$READ_KU                   00000708 RG    01
$$FOR$READ_SF                   00000050 RG    01
$$FOR$READ_SL                   00000058 RG    01
$$FOR$READ_SN                   00000730 RG    01
$$FOR$READ_SO                   00000060 RG    01
$$FOR$READ_SU                   00000068 RG    01
$$FOR$REWIND                    00000250 RG    01
$$FOR$REWRITE_SF                000006B0 RG    01
$$FOR$REWRITE_SO                000006B8 RG    01
$$FOR$REWRITE_SU                000006C0 RG    01
$$FOR$SECNDS                    00000258 RG    01
$$FOR$STOP                      00000260 RG    01
$$FOR$UNLOCK                    00000700 RG    01
$$FOR$WRITE_DF                  00000070 RG    01
$$FOR$WRITE_DO                  00000078 RG    01
$$FOR$WRITE_DU                  00000080 RG    01
$$FOR$WRITE_IF                  000006D8 RG    01
$$FOR$WRITE_IO                  000006E0 RG    01
$$FOR$WRITE_SF                  00000088 RG    01
$$FOR$WRITE_SL                  00000090 RG    01
$$FOR$WRITE_SN                  00000738 RG    01
$$FOR$WRITE_SO                  00000098 RG    01
$$FOR$WRITE_SU                  000000A0 RG    01
$$LIB$ANALYZE_SDESC             00001480 RG    01
$$LIB$ANALYZE_SDESC_R2          00001488 RG    01
$$LIB$AST_IN_PROG               000004B0 RG    01
$$LIB$ATTACH                    00000770 RG    01
$$LIB$CRC                       000004B8 RG    01
$$LIB$CRC_TABLE                 000004C0 RG    01
$$LIB$DEC_OVER                  000004C8 RG    01
$$LIB$ESTABLISH                 000004D0 RG    01
$$LIB$EXTV                      000004D8 RG    01
$$LIB$EXTZV                     000004E0 RG    01
$$LIB$FFC                       000004E8 RG    01
$$LIB$FFS                       000004F0 RG    01
$$LIB$FILE_SCAN                 000014A0 RG    01
$$LIB$FIND_FILE                 000014A8 RG    01
$$LIB$FIXUP_FLT                 000004F8 RG    01
$$LIB$FLT_UNDER                 00000500 RG    01
$$LIB$FREE_EF                   00001468 RG    01
$$LIB$FREE_LUN                  00001458 RG    01
$$LIB$FREE_VM                   000005F0 RG    01
$$LIB$GET_COMMAND               00000510 RG    01
```

```
$$LIB$GET_EF                    00001470 RG    01
$$LIB$GET_INPUT                 00000508 RG    01
$$LIB$GET_LUN                   00001460 RG    01
$$LIB$GET_OPCODE                00000780 RG    01
$$LIB$GET_VM                    000005F8 RG    01
$$LIB$INDEX                     00000518 RG    01
$$LIB$INSV                      00000520 RG    01
$$LIB$INT_OVER                  00000528 RG    01
$$LIB$LOCC                      00000530 RG    01
$$LIB$MATCHC                    00000538 RG    01
$$LIB$MATCH_COND                00000540 RG    01
$$LIB$MOVTC                     00000548 RG    01
$$LIB$MOVTUC                    00000550 RG    01
$$LIB$PUT_OUTPUT                00000558 RG    01
$$LIB$RESERVE_EF                00001478 RG    01
$$LIB$REVERT                    00000560 RG    01
$$LIB$SCANC                     00000568 RG    01
$$LIB$SCOPY_DXDX                00000570 RG    01
$$LIB$SCOPY_DXDX6               00000578 RG    01
$$LIB$SCOPY_R_DX                00000580 RG    01
$$LIB$SCOPY_R_DX6               00000588 RG    01
$$LIB$SFREE1_DD                 000005A0 RG    01
$$LIB$SFREE1_DD6                000005A8 RG    01
$$LIB$SFREEN_DD                 000005B0 RG    01
$$LIB$SFREEN_DD6                000005B8 RG    01
$$LIB$SGET1_DD                  00000590 RG    01
$$LIB$SGET1_DD_R6               00000598 RG    01
$$LIB$SHOW_VM                   00000600 RG    01
$$LIB$SIGNAL                    000005C8 RG    01
$$LIB$SIG_TO_RET                000005D8 RG    01
$$LIB$SKPC                      000005E0 RG    01
$$LIB$SPANC                     000005E8 RG    01
$$LIB$SPAWN                     00000778 RG    01
$$LIB$STAT_VM                   000005C0 RG    01
$$LIB$STOP                      000005D0 RG    01
$$LIB$TPARSE                    00000B58 RG    01
$$MTH$$AB_ALOG_V                00000F48 RG    01
$$MTH$$AB_ATAN_V                00000F50 RG    01
$$MTH$ACOS                      00000268 RG    01
$$MTH$ACOSD                     00000E00 RG    01
$$MTH$ACOSD_R4                  00000E08 RG    01
$$MTH$ACOS_R4                   00000270 RG    01
$$MTH$ACOS_R5                   00000270 RG    01
$$MTH$ALOG                      00000278 RG    01
$$MTH$ALOG10                    00000280 RG    01
$$MTH$ALOG10_R5                 00000288 RG    01
$$MTH$ALOG2                     00000F20 RG    01
$$MTH$ALOG_R5                   00000290 RG    01
$$MTH$AL_4_OV_PI_V              00000F40 RG    01
$$MTH$ASIN                      00000298 RG    01
$$MTH$ASIND                     00000E10 RG    01
$$MTH$ASIND_R4                  00000E18 RG    01
$$MTH$ASIN_R4                   000002A0 RG    01
$$MTH$ASIN_R5                   000002A0 RG    01
$$MTH$ATAN                      000002A8 RG    01
$$MTH$ATAN2                     000002B0 RG    01
$$MTH$ATAND                     00000E20 RG    01
$$MTH$ATAND2                    00000E28 RG    01
$$MTH$ATAND_R4                  00000E30 RG    01
$$MTH$ATANH                     00000ED0 RG    01
$$MTH$ATAN_R4                   000002B8 RG    01
$$MTH$CABS                      00000438 RG    01
$$MTH$CCOS                      00000458 RG    01
$$MTH$CEXP                      00000440 RG    01
$$MTH$CLOG                      00000448 RG    01
$$MTH$COS                       00000368 RG    01
$$MTH$COSD                      00000E90 RG    01
$$MTH$COSD_R4                   00000E98 RG    01
$$MTH$COSH                      00000450 RG    01
$$MTH$COS_R4                    00000370 RG    01
$$MTH$CSIN                      00000460 RG    01
$$MTH$CSQRT                     00000468 RG    01
$$MTH$DACOS                     000002C0 RG    01
$$MTH$DACOSD                    00000E38 RG    01
$$MTH$DACOSD_R7                 00000E40 RG    01
$$MTH$DACOS_R7                  000002C8 RG    01
$$MTH$DACOS_R9                  000002C8 RG    01
$$MTH$DASIN                     000002D0 RG    01
$$MTH$DASIND                    00000E48 RG    01
$$MTH$DASIND_R7                 00000E50 RG    01
$$MTH$DASIN_R7                  000002D8 RG    01
$$MTH$DASIN_R9                  000002D8 RG    01
$$MTH$DATAN                     000002E0 RG    01
$$MTH$DATAN2                    000002E8 RG    01
$$MTH$DATAND                    00000E58 RG    01
$$MTH$DATAND2                   00000E60 RG    01
$$MTH$DATAND_R7                 00000E68 RG    01
$$MTH$DATANH                    00000ED8 RG    01
$$MTH$DATAN_R7                  000002F0 RG    01
$$MTH$DCOS                      00000328 RG    01
$$MTH$DCOSD                     00000E70 RG    01
$$MTH$DCOSD_R7                  00000E78 RG    01
$$MTH$DCOSH                     00000470 RG    01
$$MTH$DCOS_R7                   00000330 RG    01
$$MTH$DEXP                      000002F8 RG    01
$$MTH$DEXP_R6                   00000300 RG    01
$$MTH$DEXP_R7                   00000300 RG    01
$$MTH$DLOG                      00000308 RG    01
$$MTH$DLOG10                    00000310 RG    01
$$MTH$DLOG10_R8                 00000318 RG    01
$$MTH$DLOG2                     00000F28 RG    01
$$MTH$DLOG_R8                   00000320 RG    01
$$MTH$DSIN                      00000338 RG    01
$$MTH$DSINCOS                   00000F00 RG    01
$$MTH$DSINCOSD                  00000F10 RG    01
$$MTH$DSINCOSD_R7               00000F18 RG    01
$$MTH$DSINCOS_R7                00000F08 RG    01
$$MTH$DSIND                     00000E80 RG    01
$$MTH$DSIND_R7                  00000E88 RG    01
$$MTH$DSINH                     00000478 RG    01
$$MTH$DSIN_R7                   00000340 RG    01
$$MTH$DSQRT                     00000348 RG    01
$$MTH$DSQRT_R5                  00000350 RG    01
$$MTH$DTAN                      00000480 RG    01
```

```
$$MTH$DTAND                     00000EB0 RG    01
$$MTH$DTAND_R7                  00000EB8 RG    01
$$MTH$DTANH                     00000488 RG    01
$$MTH$DTAN_R7                   00000710 RG    01
$$MTH$EXP                       00000358 RG    01
$$MTH$EXP_R4                    00000360 RG    01
$$MTH$RANDOM                    00000490 RG    01
$$MTH$SIN                       00000378 RG    01
$$MTH$SINCOS                    00000EE0 RG    01
$$MTH$SINCOSD                   00000EF0 RG    01
$$MTH$SINCOSD_R5                00000EF8 RG    01
$$MTH$SINCOS_R5                 00000EE8 RG    01
$$MTH$SIND                      00000EA0 RG    01
$$MTH$SIND_R4                   00000EA8 RG    01
$$MTH$SINH                      00000498 RG    01
$$MTH$SIN_R4                    00000380 RG    01
$$MTH$SQRT                      00000388 RG    01
$$MTH$SQRT_R2                   00000390 RG    01
$$MTH$SQRT_R3                   00000720 RG    01
$$MTH$TAN                       000004A0 RG    01
$$MTH$TAND                      00000EC0 RG    01
$$MTH$TAND_R4                   00000EC8 RG    01
$$MTH$TAND_R5                   00000F38 RG    01
$$MTH$TANH                      000004A8 RG    01
$$MTH$TAN_R4                    00000718 RG    01
$$MTH$TAN_R5                    00000F30 RG    01
$$OTS$$CVT_D_T_R8               00000790 RG    01
$$OTS$$CVT_G_T_R8               00000798 RG    01
$$OTS$$CVT_H_T_R8               000007A0 RG    01
$$OTS$CVT_L_TB                  00000758 RG    01
$$OTS$CVT_L_TI                  00000690 RG    01
$$OTS$CVT_L_TL                  000006A8 RG    01
$$OTS$CVT_L_TO                  00000698 RG    01
$$OTS$CVT_L_TZ                  000006A0 RG    01
$$OTS$CVT_TB_L                  00000760 RG    01
$$OTS$CVT_TI_L                  00000210 RG    01
$$OTS$CVT_TL_L                  00000218 RG    01
$$OTS$CVT_TO_L                  00000220 RG    01
$$OTS$CVT_TZ_L                  00000228 RG    01
$$OTS$CVT_T_D                   00000200 RG    01
$$OTS$CVT_T_F                   00000768 RG    01
$$OTS$CVT_T_G                   00000660 RG    01
$$OTS$CVT_T_H                   00000688 RG    01
$$OTS$DIVC                      00000398 RG    01
$$OTS$POWCJ                     000003A0 RG    01
$$OTS$POWDD                     000003A8 RG    01
$$OTS$POWDJ                     000003C0 RG    01
$$OTS$POWDR                     000003B0 RG    01
$$OTS$POWII                     000003C8 RG    01
$$OTS$POWJJ                     000003D0 RG    01
$$OTS$POWRD                     000003B8 RG    01
$$OTS$POWRJ                     000003D8 RG    01
$$OTS$POWRR                     000003E0 RG    01
$$OTS$$SCOPY_DXDX               000003E8 RG    01
$$OTS$$SCOPY_DXDX6              000003F0 RG    01
$$OTS$$SCOPY_R_DX               000003F8 RG    01
$$OTS$$SCOPY_R_DX6              00000400 RG    01
$$OTS$$FREE1_DD                 00000418 RG    01
$$OTS$$FREE1_DD6                00000420 RG    01
$$OTS$$FREEN_DD                 00000428 RG    01
$$OTS$$FREEN_DD6                00000430 RG    01
$$OTS$$GET1_DD                  00000408 RG    01
$$OTS$$GET1_DD_R6               00000410 RG    01
$$STR$ANALYZE_SDESC             00001490 RG    01
$$STR$ANALYZE_SDESC_R1          00001498 RG    01
$$STR$APPEND                    00000C60 RG    01
$$STR$COMPARE                   00000C68 RG    01
$$STR$COMPARE_EQL               00000C70 RG    01
$$STR$CONCAT                    00000800 RG    01
$$STR$COPY_DX                   00000808 RG    01
$$STR$COPY_DX_R8                00000C10 RG    01
$$STR$COPY_R                    00000810 RG    01
$$STR$COPY_R_R8                 00000C18 RG    01
$$STR$DUPL_CHAR                 00000850 RG    01
$$STR$DUPL_CHARR8               00000C20 RG    01
$$STR$FREE1_DX                  00000818 RG    01
$$STR$FREE1_DX_R4               00000C28 RG    01
$$STR$GET1_DX                   00000820 RG    01
$$STR$GET1_DX_R4                00000C30 RG    01
$$STR$LEFT                      00000828 RG    01
$$STR$LEFT_R8                   00000C38 RG    01
$$STR$LEN_EXTR                  00000830 RG    01
$$STR$LEN_EXTR_R8               00000C40 RG    01
$$STR$POSITION                  00000840 RG    01
$$STR$POSITION_R6               00000C48 RG    01
$$STR$POS_EXTR                  00000838 RG    01
$$STR$POS_EXTR_R8               00000C50 RG    01
$$STR$PREFIX                    00000C78 RG    01
$$STR$REPLACE                   00000C80 RG    01
$$STR$REPLACE_R8                00000C88 RG    01
$$STR$RIGHT                     00000848 RG    01
$$STR$RIGHT_R8                  00000C58 RG    01
$$STR$TRANSLATE                 00000C90 RG    01
$$STR$TRIM                      00000858 RG    01
$$STR$UPCASE                    00000C98 RG    01
BAS$$BLNK_LINE                  ********    X  01
BAS$$CB_GET                     ********    X  01
BAS$$CB_POP                     ********    X  01
BAS$$CB_PUSH                    ********    X  01
BAS$$CLOSE_ALL                  ********    X  01
BAS$$CTRLC_INIT                 ********    X  01
BAS$$ERR_INIT                   ********    X  01
BAS$$FORMAT_INT                 ********    X  01
BAS$$HANDLER                    ********    X  01
BAS$$NEXT_LUN                   ********    X  01
BAS$$OPEN_ZERO                  ********    X  01
BAS$$RECOD_INIT                 ********    X  01
BAS$$REC_WSL1                   ********    X  01
BAS$$SCALE_L_R1                 ********    X  01
BAS$$SCALE_RT                   ********    X  01
BAS$$SIGNAL                     ********    X  01
BAS$$SIGNAL_IO                  ********    X  01
BAS$$STATU_INIT                 ********    X  01
BAS$$STOP                       ********    X  01
```

BAS$$STOP_IO	********	X	01	BAS$INIT_R8	********	X	01				
BAS$$STOP_RMS	********	X	01	BAS$INPUT	********	X	01				
BAS$$UDF_RL1	********	X	01	BAS$INPUT_LINE	********	X	01				
BAS$$UDF_WL1	********	X	01	BAS$INSTR	********	X	01				
BAS$ANSI_INPUT	********	X	01	BAS$IN_B_R	********	X	01				
BAS$ANSI_IO_END	********	X	01	BAS$IN_D_R	********	X	01				
BAS$BUFSIZ	********	X	01	BAS$IN_F_R	********	X	01				
BAS$CANTYPAHEAD	********	X	01	BAS$IN_G_R	********	X	01				
BAS$CCPOS	********	X	01	BAS$IN_H_R	********	X	01				
BAS$CHR	********	X	01	BAS$IN_L_R	********	X	01				
BAS$CLOSE	********	X	01	BAS$IN_P_DX	********	X	01				
BAS$CMPD_APP	********	X	01	BAS$IN_T_DX	********	X	01				
BAS$CMPF_APP	********	X	01	BAS$IN_W_R	********	X	01				
BAS$CMPG_APP	********	X	01	BAS$IO_END	********	X	01				
BAS$CMPH_APP	********	X	01	BAS$LINPUT	********	X	01				
BAS$CTRLC	********	X	01	BAS$MAT_INPUT	********	X	01				
BAS$CVT_OUT_D_E	********	X	01	BAS$MAT_LINPUT	********	X	01				
BAS$CVT_OUT_D_F	********	X	01	BAS$MAT_PRINT	********	X	01				
BAS$CVT_OUT_D_G	********	X	01	BAS$MAT_READ	********	X	01				
BAS$CVT_OUT_F_E	********	X	01	BAS$NUMT_D	********	X	01				
BAS$CVT_OUT_F_F	********	X	01	BAS$NUMT_F	********	X	01				
BAS$CVT_OUT_G_E	********	X	01	BAS$NUMT_G	********	X	01				
BAS$CVT_OUT_G_F	********	X	01	BAS$NUMT_H	********	X	01				
BAS$CVT_OUT_G_G	********	X	01	BAS$NUMT_L	********	X	01				
BAS$CVT_OUT_H_E	********	X	01	BAS$NUMT_P	********	X	01				
BAS$CVT_OUT_H_F	********	X	01	BAS$NUM_D	********	X	01				
BAS$CVT_OUT_H_G	********	X	01	BAS$NUM_F	********	X	01				
BAS$CVT_OUT_P_E	********	X	01	BAS$NUM_G	********	X	01				
BAS$CVT_OUT_P_F	********	X	01	BAS$NUM_H	********	X	01				
BAS$CVT_OUT_P_G	********	X	01	BAS$NUM_L	********	X	01				
BAS$CVT_T_P	********	X	01	BAS$NUM_P	********	X	01				
BAS$DELETE	********	X	01	BAS$ON_ERR_BK	********	X	01				
BAS$DSCALE_D_R1	********	X	01	BAS$ON_ERR_Z	********	X	01				
BAS$EDIT	********	X	01	BAS$OPEN	********	X	01				
BAS$END_DEF_R8	********	X	01	BAS$OUT_D_V_B	********	X	01				
BAS$END_DFS_R8	********	X	01	BAS$OUT_D_V_C	********	X	01				
BAS$END_GSB_R8	********	X	01	BAS$OUT_D_V_S	********	X	01				
BAS$END_R8	********	X	01	BAS$OUT_F_V_B	********	X	01				
BAS$ERL	********	X	01	BAS$OUT_F_V_C	********	X	01				
BAS$ERN	********	X	01	BAS$OUT_F_V_S	********	X	01				
BAS$ERR	********	X	01	BAS$OUT_G_V_B	********	X	01				
BAS$ERROR	********	X	01	BAS$OUT_G_V_C	********	X	01				
BAS$ERT	********	X	01	BAS$OUT_G_V_S	********	X	01				
BAS$FIND	********	X	01	BAS$OUT_H_V_B	********	X	01				
BAS$FIND_KEY	********	X	01	BAS$OUT_H_V_C	********	X	01				
BAS$FIND_RECORD	********	X	01	BAS$OUT_H_V_S	********	X	01				
BAS$FIND_RFA	********	X	01	BAS$OUT_L_V_B	********	X	01				
BAS$FREE	********	X	01	BAS$OUT_L_V_C	********	X	01				
BAS$GET	********	X	01	BAS$OUT_L_V_S	********	X	01				
BAS$GETRFA	********	X	01	BAS$OUT_P_DX_B	********	X	01				
BAS$GET_KEY	********	X	01	BAS$OUT_P_DX_C	********	X	01				
BAS$GET_RECORD	********	X	01	BAS$OUT_P_DX_S	********	X	01				
BAS$GET_RFA	********	X	01	BAS$OUT_T_DX_B	********	X	01				
BAS$HANDLER	********	X	01	BAS$OUT_T_DX_C	********	X	01				
BAS$INIT_DEF_R8	********	X	01	BAS$OUT_T_DX_S	********	X	01				
BAS$INIT_DFS_R8	********	X	01	BAS$POP_ERR	********	X	01				
BAS$INIT_GOSUB	********	X	01	BAS$PRINT	********	X	01				

BAS$PRINT_USING	********	X	01	COB$CVTRIP_R9	********	X	01
BAS$PUSH_ERR	********	X	01	COB$CVTRIQ_R8	********	X	01
BAS$PUT	********	X	01	COB$CVTRIW_R8	********	X	01
BAS$PUT_COUNT	********	X	01	COB$CVTRPQ_R9	********	X	01
BAS$PUT_RECORD	********	X	01	COB$CVTRQP_R9	********	X	01
BAS$PUT_REC_CNT	********	X	01	COB$CVTTI_R8	********	X	01
BAS$RCTRLC	********	X	01	COB$CVTWI_R8	********	X	01
BAS$READ	********	X	01	COB$DISPLAY	********	X	01
BAS$RECOUNT	********	X	01	COB$DISP_NO_ADV	********	X	01
BAS$RESTORE	********	X	01	COB$DIVI	********	X	01
BAS$RESTORE_DAT	********	X	01	COB$DIVI_OSE	********	X	01
BAS$RESTORE_KEY	********	X	01	COB$DIVQ_R8	********	X	01
BAS$RESUME	********	X	01	COB$ERROR	********	X	01
BAS$RESUME_Z	********	X	01	COB$HANDLER	********	X	01
BAS$RSET	********	X	01	COB$IOEXCEPTION	********	X	01
BAS$RSET_R	********	X	01	COB$MULI	********	X	01
BAS$SCALE_D_R1	********	X	01	COB$MULQ_R8	********	X	01
BAS$SCRATCH	********	X	01	COB$PAUSE	********	X	01
BAS$STATUS	********	X	01	COB$SUBI	********	X	01
BAS$STR_D	********	X	01	FOR$$CB_GET	********	X	01
BAS$STR_F	********	X	01	FOR$$CB_POP	********	X	01
BAS$STR_G	********	X	01	FOR$$CB_PUSH	********	X	01
BAS$STR_H	********	X	01	FOR$$CB_RET	********	X	01
BAS$STR_L	********	X	01	FOR$$ERRSNS_SAV	********	X	01
BAS$STR_P	********	X	01	FOR$$FP_MATCH	********	X	01
BAS$UNLOCK	********	X	01	FOR$$IO_BEG	********	X	01
BAS$UPDATE	********	X	01	FOR$BACKSPACE	********	X	01
BAS$UPDATE_COUN	********	X	01	FOR$CLOSE	********	X	01
BAS$VAL_D	********	X	01	FOR$CNV_OUT_I	********	X	01
BAS$VAL_F	********	X	01	FOR$CNV_OUT_L	********	X	01
BAS$VAL_G	********	X	01	FOR$CNV_OUT_O	********	X	01
BAS$VAL_H	********	X	01	FOR$CNV_OUT_Z	********	X	01
BAS$VAL_L	********	X	01	FOR$CVT_D_TD	********	X	01
BAS$VAL_P	********	X	01	FOR$CVT_D_TE	********	X	01
BAS$WAIT	********	X	01	FOR$CVT_D_TF	********	X	01
COB$$HANDLER	********	X	01	FOR$CVT_D_TG	********	X	01
COB$ACCEPT	********	X	01	FOR$CVT_G_TD	********	X	01
COB$ACC_DATE	********	X	01	FOR$CVT_G_TE	********	X	01
COB$ACC_DAY	********	X	01	FOR$CVT_G_TF	********	X	01
COB$ACC_DAYWEEK	********	X	01	FOR$CVT_G_TG	********	X	01
COB$ACC_TIME	********	X	01	FOR$CVT_H_TD	********	X	01
COB$ADDI	********	X	01	FOR$CVT_H_TE	********	X	01
COB$CMPI	********	X	01	FOR$CVT_H_TF	********	X	01
COB$CVTDI_R7	********	X	01	FOR$CVT_H_TG	********	X	01
COB$CVTFI_R7	********	X	01	FOR$DECODE_MF	********	X	01
COB$CVTID_R7	********	X	01	FOR$DECODE_MO	********	X	01
COB$CVTIF_R7	********	X	01	FOR$DEF_FILE	********	X	01
COB$CVTIL_R8	********	X	01	FOR$DEF_FILE_W	********	X	01
COB$CVTIP_R9	********	X	01	FOR$DELETE	********	X	01
COB$CVTIQ_R8	********	X	01	FOR$DELETE_D	********	X	01
COB$CVTIW_R8	********	X	01	FOR$ENCODE_MF	********	X	01
COB$CVTLI_R8	********	X	01	FOR$ENCODE_MO	********	X	01
COB$CVTPI_R9	********	X	01	FOR$ENDFILE	********	X	01
COB$CVTPQ_R9	********	X	01	FOR$ERRSNS	********	X	01
COB$CVTQI_R8	********	X	01	FOR$ERRSNS_W	********	X	01
COB$CVTQP_R9	********	X	01	FOR$EXIT	********	X	01
COB$CVTRIC_R8	********	X	01	FOR$EXIT_W	********	X	01

FOR$FIND	********	X	01	FOR$STOP	********	X	01				
FOR$INI_DES1_R2	********	X	01	FOR$UNLOCK	********	X	01				
FOR$INI_DES2_R3	********	X	01	FOR$WRITE_DF	********	X	01				
FOR$INI_DESC_R6	********	X	01	FOR$WRITE_DO	********	X	01				
FOR$INQUIRE	********	X	01	FOR$WRITE_DU	********	X	01				
FOR$IO_B_R	********	X	01	FOR$WRITE_IF	********	X	01				
FOR$IO_B_V	********	X	01	FOR$WRITE_IO	********	X	01				
FOR$IO_DC_R	********	X	01	FOR$WRITE_SF	********	X	01				
FOR$IO_DC_V	********	X	01	FOR$WRITE_SL	********	X	01				
FOR$IO_D_R	********	X	01	FOR$WRITE_SN	********	X	01				
FOR$IO_D_V	********	X	01	FOR$WRITE_SO	********	X	01				
FOR$IO_END	********	X	01	FOR$WRITE_SU	********	X	01				
FOR$IO_FC_R	********	X	01	LIB$ANALYZE_SDESC	********	X	01				
FOR$IO_FC_V	********	X	01	LIB$ANALYZE_SDESC_R2	********	X	01				
FOR$IO_F_R	********	X	01	LIB$AST_IN_PROG	********	X	01				
FOR$IO_F_V	********	X	01	LIB$ATTACH	********	X	01				
FOR$IO_GC_R	********	X	01	LIB$CRC	********	X	01				
FOR$IO_GC_V	********	X	01	LIB$CRC_TABLE	********	X	01				
FOR$IO_G_R	********	X	01	LIB$DEC_OVER	********	X	01				
FOR$IO_G_V	********	X	01	LIB$ESTABLISH	********	X	01				
FOR$IO_H_R	********	X	01	LIB$EXTV	********	X	01				
FOR$IO_H_V	********	X	01	LIB$EXTZV	********	X	01				
FOR$IO_LO_R	********	X	01	LIB$FFC	********	X	01				
FOR$IO_LU_V	********	X	01	LIB$FFS	********	X	01				
FOR$IO_L_R	********	X	01	LIB$FILE_SCAN	********	X	01				
FOR$IO_L_V	********	X	01	LIB$FIND_FILE	********	X	01				
FOR$IO_T_DS	********	X	01	LIB$FIXUP_FLT	********	X	01				
FOR$IO_T_V_DS	********	X	01	LIB$FLT_UNDER	********	X	01				
FOR$IO_WO_R	********	X	01	LIB$FREE_EF	********	X	01				
FOR$IO_WU_V	********	X	01	LIB$FREE_LUN	********	X	01				
FOR$IO_W_R	********	X	01	LIB$FREE_VM	********	X	01				
FOR$IO_W_V	********	X	01	LIB$GET_COMMAND	********	X	01				
FOR$IO_X_DA	********	X	01	LIB$GET_EF	********	X	01				
FOR$IO_X_NL	********	X	01	LIB$GET_INPUT	********	X	01				
FOR$IO_X_SB	********	X	01	LIB$GET_LUN	********	X	01				
FOR$IO_X_SE	********	X	01	LIB$GET_OPCODE	********	X	01				
FOR$OPEN	********	X	01	LIB$GET_VM	********	X	01				
FOR$PAUSE	********	X	01	LIB$INDEX	********	X	01				
FOR$RAB	********	X	01	LIB$INSV	********	X	01				
FOR$READ_DF	********	X	01	LIB$INT_OVER	********	X	01				
FOR$READ_DO	********	X	01	LIB$LOCC	********	X	01				
FOR$READ_DU	********	X	01	LIB$MATCHC	********	X	01				
FOR$READ_IF	********	X	01	LIB$MATCH_COND	********	X	01				
FOR$READ_IO	********	X	01	LIB$MOVTC	********	X	01				
FOR$READ_KF	********	X	01	LIB$MOVTUC	********	X	01				
FOR$READ_KO	********	X	01	LIB$PUT_OUTPUT	********	X	01				
FOR$READ_KU	********	X	01	LIB$RESERVE_EF	********	X	01				
FOR$READ_SF	********	X	01	LIB$REVERT	********	X	01				
FOR$READ_SL	********	X	01	LIB$SCANC	********	X	01				
FOR$READ_SN	********	X	01	LIB$SCOPY_DXDX	********	X	01				
FOR$READ_SO	********	X	01	LIB$SCOPY_DXDX6	********	X	01				
FOR$READ_SU	********	X	01	LIB$SCOPY_R_DX	********	X	01				
FOR$REWIND	********	X	01	LIB$SCOPY_R_DX6	********	X	01				
FOR$REWRITE_SF	********	X	01	LIB$SFREE1_DD	********	X	01				
FOR$REWRITE_SO	********	X	01	LIB$SFREE1_DD6	********	X	01				
FOR$REWRITE_SU	********	X	01	LIB$SFREEN_DD	********	X	01				
FOR$SECNDS	********	X	01	LIB$SFREEN_DD6	********	X	01				

LIB$SGET1_DD	********	X	01	MTH$DATAND_R7	********	X	01
LIB$SGET1_DD_R6	********	X	01	MTH$DATANH	********	X	01
LIB$SHOW_VM	********	X	01	MTH$DATAN_R7	********	X	01
LIB$SIGNAL	********	X	01	MTH$DCOS	********	X	01
LIB$SIG_TO_RET	********	X	01	MTH$DCOSD	********	X	01
LIB$SKPC	********	X	01	MTH$DCOSD_R7	********	X	01
LIB$SPANC	********	X	01	MTH$DCOSH	********	X	01
LIB$SPAWN	********	X	01	MTH$DCOS_R7	********	X	01
LIB$STAT_VM	********	X	01	MTH$DEXP	********	X	01
LIB$STOP	********	X	01	MTH$DEXP_R6	********	X	01
LIB$TPARSE	********	X	01	MTH$DLOG	********	X	01
MTH$$AB_ALOG	00001650 R		01	MTH$DLOG10	********	X	01
MTH$$AB_ATAN	00001750 R		01	MTH$DLOG10_R8	********	X	01
MTH$ACOS	********	X	01	MTH$DLOG2	********	X	01
MTH$ACOSD	********	X	01	MTH$DLOG_R8	********	X	01
MTH$ACOSD_R4	********	X	01	MTH$DSIN	********	X	01
MTH$ACOS_R4	********	X	01	MTH$DSINCOS	********	X	01
MTH$ALOG	********	X	01	MTH$DSINCOSD	********	X	01
MTH$ALOG10	********	X	01	MTH$DSINCOSD_R7	********	X	01
MTH$ALOG10_R5	********	X	01	MTH$DSINCOS_R7	********	X	01
MTH$ALOG2	********	X	01	MTH$DSIND	********	X	01
MTH$ALOG_R5	********	X	01	MTH$DSIND_R7	********	X	01
MTH$AL_4_OV_PI	********	X	01	MTH$DSINH	********	X	01
MTH$ASIN	********	X	01	MTH$DSIN_R7	********	X	01
MTH$ASIND	********	X	01	MTH$DSQRT	********	X	01
MTH$ASIND_R4	********	X	01	MTH$DSQRT_R5	********	X	01
MTH$ASIN_R4	********	X	01	MTH$DTAN	********	X	01
MTH$ATAN	********	X	01	MTH$DTAND	********	X	01
MTH$ATAN2	********	X	01	MTH$DTAND_R7	********	X	01
MTH$ATAND	********	X	01	MTH$DTANH	********	X	01
MTH$ATAND2	********	X	01	MTH$DTAN_R7	********	X	01
MTH$ATAND_R4	********	X	01	MTH$EXP	********	X	01
MTH$ATANH	********	X	01	MTH$EXP_R4	********	X	01
MTH$ATAN_R4	********	X	01	MTH$RANDOM	********	X	01
MTH$CABS	********	X	01	MTH$SIN	********	X	01
MTH$CCOS	********	X	01	MTH$SINCOS	********	X	01
MTH$CEXP	********	X	01	MTH$SINCOSD	********	X	01
MTH$CLOG	********	X	01	MTH$SINCOSD_R5	********	X	01
MTH$COS	********	X	01	MTH$SINCOS_R5	********	X	01
MTH$COSD	********	X	01	MTH$SIND	********	X	01
MTH$COSD_R4	********	X	01	MTH$SIND_R4	********	X	01
MTH$COSH	********	X	01	MTH$SINH	********	X	01
MTH$COS_R4	********	X	01	MTH$SIN_R4	********	X	01
MTH$CSIN	********	X	01	MTH$SQRT	********	X	01
MTH$CSQRT	********	X	01	MTH$SQRT_R2	********	X	01
MTH$DACOS	********	X	01	MTH$SQRT_R3	********	X	01
MTH$DACOSD	********	X	01	MTH$TAN	********	X	01
MTH$DACOSD_R7	********	X	01	MTH$TAND	********	X	01
MTH$DACOS_R7	********	X	01	MTH$TAND_R4	********	X	01
MTH$DASIN	********	X	01	MTH$TAND_R5	********	X	01
MTH$DASIND	********	X	01	MTH$TANH	********	X	01
MTH$DASIND_R7	********	X	01	MTH$TAN_R4	********	X	01
MTH$DASIN_R7	********	X	01	MTH$TAN_R5	********	X	01
MTH$DATAN	********	X	01	OTS$$CVT_D_T_R8	********	X	01
MTH$DATAN2	********	X	01	OTS$$CVT_G_T_R8	********	X	01
MTH$DATAND	********	X	01	OTS$$CVT_H_T_R8	********	X	01
MTH$DATAND2	********	X	01	OTS$CVT_L_TB	********	X	01

OTS$CVT_L_TI	********	X	01	STR$POS_EXTR_R8	********	X	01
OTS$CVT_L_TL	********	X	01	STR$PREFIX	********	X	01
OTS$CVT_L_TO	********	X	01	STR$REPLACE	********	X	01
OTS$CVT_L_TZ	********	X	01	STR$REPLACE_R8	********	X	01
OTS$CVT_TB_L	********	X	01	STR$RIGHT	********	X	01
OTS$CVT_TI_L	********	X	01	STR$RIGHT_R8	********	X	01
OTS$CVT_TL_L	********	X	01	STR$TRANSLATE	********	X	01
OTS$CVT_TO_L	********	X	01	STR$TRIM	********	X	01
OTS$CVT_TZ_L	********	X	01	STR$UPCASE	********	X	01
OTS$CVT_T_D	********	X	01				
OTS$CVT_T_F	********	X	01				
OTS$CVT_T_G	********	X	01				
OTS$CVT_T_H	********	X	01				
OTS$DIVC	********	X	01				
OTS$POWCJ	********	X	01				
OTS$POWDD	********	X	01				
OTS$POWDJ	********	X	01				
OTS$POWDR	********	X	01				
OTS$POWII	********	X	01				
OTS$POWJJ	********	X	01				
OTS$POWRD	********	X	01				
OTS$POWRJ	********	X	01				
OTS$POWRR	********	X	01				
OTS$SCOPY_DXDX	********	X	01				
OTS$SCOPY_DXDX6	********	X	01				
OTS$SCOPY_R_DX	********	X	01				
OTS$SCOPY_R_DX6	********	X	01				
OTS$SFREE1_DD	********	X	01				
OTS$SFREE1_DD6	********	X	01				
OTS$SFREEN_DD	********	X	01				
OTS$SFREEN_DD6	********	X	01				
OTS$SGET1_DD	********	X	01				
OTS$SGET1_DD_R6	********	X	01				
RTL$START	00000000	R	01				
STR$ANALYZE_SDESC	********	X	01				
STR$ANALYZE_SDESC_R1	********	X	01				
STR$APPEND	********	X	01				
STR$COMPARE	********	X	01				
STR$COMPARE_EQL	********	X	01				
STR$CONCAT	********	X	01				
STR$COPY_DX	********	X	01				
STR$COPY_DX_R8	********	X	01				
STR$COPY_R	********	X	01				
STR$COPY_R_R8	********	X	01				
STR$DUPL_CHAR	********	X	01				
STR$DUPL_CHARR8	********	X	01				
STR$FREE1_DX	********	X	01				
STR$FREE1_DX_R4	********	X	01				
STR$GET1_DX	********	X	01				
STR$GET1_DX_R4	********	X	01				
STR$LEFT	********	X	01				
STR$LEFT_R8	********	X	01				
STR$LEN_EXTR	********	X	01				
STR$LEN_EXTR_R8	********	X	01				
STR$POSITION	********	X	01				
STR$POSITION_R6	********	X	01				
STR$POS_EXTR	********	X	01				

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
VMSVECTOR	00001787 (6023.)	01 (1.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC PAGE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.72
Command processing	135	00:00:00.60	00:00:03.85
Pass 1	799	00:01:03.40	00:01:50.18
Symbol table sort	0	00:00:01.57	00:00:02.25
Pass 2	475	00:00:17.00	00:00:41.89
Symbol table output	1	00:00:00.79	00:00:01.75
Psect synopsis output	0	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	1441	00:01:23.48	00:02:40.68

The working set limit was 2400 pages.
319752 bytes (625 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1092 non-local and 0 local symbols.
3082 source lines were read in Pass 1, producing 93 object records in Pass 2.
3 pages of virtual memory were used to define 2 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/LIS=LIS$:VMSVECTOR/OBJ=OBJ$:VMSVECTOR MSRC$:VMSVECTOR/UPDATE=(ENH$:VMSVECTOR)

VMSVECTOR
LIS

VMSRTL

VMSRTL
MAP